



Centro Universitário de Brasília – Uniceub
Faculdade de Ciências Sociais e Tecnologia Aplicadas - FATECS

Monitor de Distância de Segurança para Automóveis Utilizando Sensor de Distância

Aluno: Gustavo e Silva Maciel – RA: 2021819-0

Orientador: Prof. Thiago de Miranda Leão Toribio

Brasília – DF, Junho de 2009

GUSTAVO E SILVA MACIEL

**Monitor de Distância de Segurança para
Automóveis Utilizando Sensor de Distância**

Trabalho de conclusão de curso
apresentado como parte das
atividades para obtenção do título de
Bacharelado em Engenharia de
Computação, do curso de
Engenharia de Computação da
Faculdade de Tecnologia e Ciências
Sociais Aplicadas de Brasília –
UniCEUB.

Profº Orientador: Professor M.C. Thiago De Miranda Leão Toribio
Brasília – DF, 1º semestre de 2009

Autoria: Gustavo e Silva Maciel

Título: Monitor de Distância de Segurança para Automóveis utilizando Sensor de Distância

Trabalho de conclusão de curso apresentado como parte das atividades para obtenção do título de Bacharelado em Engenharia de Computação, do curso de Engenharia de Computação da Faculdade de Tecnologia e Ciências Sociais Aplicadas do Centro Universitário de Brasília – UniCEUB.

Os componentes da banca de avaliação, abaixo listados, consideram este trabalho aprovado.				
	Nome	Titulação	Assinatura	Instituição
1				
2				
3				

Data da aprovação: ____ de _____ de _____.

DEDICATÓRIA

Aos meus pais e meu irmão.

AGRADECIMENTO

Agradeço, primeiramente, a Deus que me deu força e coragem para concluir este trabalho. Agradeço aos meus pais, Nivaldo e Ana Cristina, meu irmão Tales e minha madrinha Iraneide, que me deram todo apoio nos momentos de dificuldades para conclusão deste projeto. Agradeço meu orientador Thiago Toribio que me auxiliou e guiou corretamente no desenvolvimento desta obra. Agradeço meu padrinho Geraldo Majela que ajudou na montagem da plataforma utilizada neste projeto. Agradeço o colega Mario Gomide me auxiliou no projeto. Agradeço a presença dos amigos na minha apresentação, Marcus Felipe, Leonardo Vítor e José Jackson e, também, da minha prima Daniella, bem como os demais. E fico grato pelo auxílio dos demais colegas e professores que me acompanharam neste curso de Engenharia de Computação ao longo desses anos.

Resumo

O projeto desta monografia de Engenharia de Computação é o desenvolvimento de um dispositivo que auxilia o motorista a manter a distância segura entre seu veículo e o que segue a frente. Desta forma, o motorista terá espaço suficiente para reagir a tempo de evitar um acidente frontal de seu veículo, ou pelo menos tentar reduzir os danos se este acontecer. Este estudo traz uma análise da distância segura que o motorista deve ter em relação ao carro que segue, levando em consideração, sobretudo, a velocidade atual do veículo.

Com base na distância segura definida é desenvolvido um protótipo que alerta o motorista da distância correta que se deve manter do automóvel que segue a frente. É utilizado um kit de desenvolvimento com microcontrolador PIC 16F877A, um sensor de distância ultra-sônico com alcance de 6m, um LED indicador de distância segura e um alarme sonoro de alerta para manter distância. A programação do protótipo é desenvolvida em linguagem de programação C. A proposta de apresentação deste projeto inclui a demonstração do protótipo em escala reduzida de um automóvel real simulando a velocidade em que o veículo encontra-se, considerando que o limite de sensibilidade do sensor que não se aplica ao veículo real.

Palavras-chave: *Distância segura, acidente de trânsito, sensor de distância, microcontrolador, PIC 16F877A*

Abstract

The project of this Computer Engineering paper is the development of a device that assists drivers to maintain a safe distance between their car and the car in front of them. Therefore, the driver will have enough space to react in time to avoid a frontal collision, or at least try to reduce the damage on the event of the crash. This study shows an analysis of safe distance the driver must be from the car ahead, considering overall, the current vehicle speed.

Based on the established safe distance, a prototype is built to alert the driver with the correct distance to be kept from the automobile up front. It is built with a development kit with the PIC 16F877A microcontroller, a 6 meter range sonar sensor, a safe distance indicator LED and a sound alarm to keep distance. The prototype is developed in programming language C. The proposed project presentation includes a reduced scale prototype demonstration of a real automobile simulating the current speed of this vehicle, considering the sensitivity of the sensor, that does not apply to a real vehicle.

Keywords: Safe distance, traffic accident, proximity sensor, microcontroller, PIC 16F877A

SUMÁRIO

Monitor de Distância para Automóveis para Prevenção de Acidentes utilizando Sensor de Distância

Lista de Figuras.....	XI
Lista de Tabelas	XIII
Lista de Siglas	XIV
Capítulo 1 – Introdução	1
1.1 - Motivação	2
1.2 Objetivos.....	4
1.3 Metodologia	5
1.4 - Estrutura da Monografia	6
Capítulo 2 – Definições de distância de segurança frontal entre veículos.....	8
2.1 – Direção Defensiva - Distância segura.....	8
2.2 – Dinâmica de frenagem e distância de visibilidade de parada ...	13
2.2.1 - Conceitos físicos da frenagem.....	13
2.2.2 - Distância de visibilidade de parada.....	15
2.3 - Estudo comparativo entre os métodos de cálculo de distância segura	19
Capítulo 3 – Sensor de distância ultra-sônico MaxBotix EZ1™	23
3.1 – Princípio de funcionamento do sensor ultra-sônico	24

3.2 - Sensor ultra-sônico Maxbotix® MaxSonar® - EZ1™ -	
Características e motivação de uso	26
3.2.1 – Características gerais.....	26
3.2.2 – Motivação de uso do sensor Maxbotix EZ1.....	27
3.2.3 - Dimensões e descrição das saídas de dados do LV -	
MaxSonar - EZ1™	29
Capítulo 4 – Kit de desenvolvimento MultiPIC.....	32
4.1 – Descrição do Kit MultiPIC	32
4.1.1 – Display de LCD	36
4.2 –Características do Microcontrolador PIC 16F877A.....	37
4.2.1 – Interrupções do PIC 16F877A.....	40
4.2.2 – Conversor A/D do PIC 16F877A	42
Capítulo 5 – Implementação.....	44
5.1 – Descrição do protótipo.....	44
5.2 – Ambiente de simulação do protótipo.....	50
5.3 – Programação do kit de desenvolvimento.....	51
5.3.1 – Principais partes do algoritmo	52
5.4 – Dificuldades apresentadas	54
5.4.1 – Dificuldades no algoritmo	55
5.4.2 – Problemas com o sensor de distância e kit de	
desenvolvimento	56

Capítulo 6 – Testes e Resultados	57
6.1 – Testes gerais do sensor	57
6.2 – Calibragem do sensor.....	60
6.3 – Teste do alerta de distância segura.....	62
Capítulo 7 – Conclusão	64
7.1 – Conclusões gerais	64
7.2 – Conclusões dos métodos de distância segura	65
7.3 – Conclusões de aspecto técnico	65
7.4 – Propostas de trabalhos futuros.....	66
Referência Bibliográfica.....	68
Apêndice 1 – Algoritmo do Projeto	72
Apêndice 2 – Diagrama elétrico do protótipo.....	80
Apêndice 3 – Custo do protótipo	81

Lista de Figuras

FIGURA 1 – DISTÂNCIA PERCORRIDA A 80 KM/H	9
FIGURA 2 - CONSEQUÊNCIA DE COLISÃO DE VEÍCULO DEPOIS DE FREADA BRUSCA (FONTE: G1.COM)	10
FIGURA 3 - DIAGRAMA DA REGRA DOS DOIS SEGUNDOS	12
FIGURA 4 - DISTÂNCIA DE VISIBILIDADE DE PARADA (FONTE: TOPOGRAFIA GERAL) ...	16
FIGURA 5 – TRANSMISSÃO E RECEPÇÃO DO ECO ULTRA-SÔNICO	25
FIGURA 6 - SENSOR MAXSENSOR EZ1™ DIMENSÕES (FONTE: MAX SONAR, 2008)	29
FIGURA 7- DIAGRAMA DO CIRCUITO ELÉTRICO DO SENSOR MAX SENSOR EZ1 (FONTE: MAX SONAR, 2008).....	31
FIGURA 8 - KIT DE DESENVOLVIMENTO MULTIPIC (FONTE: SMARTRADIO, 2008).....	34
FIGURA 9 - DIAGRAMA DO CIRCUITO DO MULTIPIC (FONTE: SMARTRADIO, 2008)	35
FIGURA 10 - MODELO DE AUTOMÓVEL COM SENSOR FIXADO A FRENTE	44
FIGURA 11 - FLUXOGRAMA DO PROTÓTIPO	46
FIGURA 12 – MULTIPIC EM FUNCIONAMENTO E LOCALIZAÇÃO DOS RECURSOS	47
FIGURA 13 - PROTÓTIPO MONTADO E FUNCIONANDO	49
FIGURA 14 - PROTÓTIPO SENDO TESTADO NA PLATAFORMA.....	51
FIGURA 15 - TESTE DO SENSOR TENSÃO MÍNIMA DE SAÍDA NO PINO AN	58
FIGURA 16 - FILTRO PARA REDUÇÃO DE RUÍDOS DO SENSOR	59
FIGURA 17 - TESTE DA DISTÂNCIA MEDIDA COM DESTAQUE PARA A DISTÂNCIA MEDIDA	60

FIGURA 18 - TESTE INDICANDO QUE A DISTÂNCIA É INSEGURA.....	63
---	----

Lista de Tabelas

TABELA 1 - TIPOS DE DISTRAÇÃO E PERCENTUAL DE OCORRÊNCIA [AAA FOUNDATION, 2001].....	3
TABELA 2 - COEFICIENTE DE ATRITO EM ALGUNS TIPOS DE PAVIMENTO [ARAGÃO, 2003]	18
TABELA 3 - DISTÂNCIA DE SEGURANÇA USANDO A "REGRAS DOS DOIS SEGUNDOS"	19
TABELA 4 - DISTÂNCIA DE SEGURANÇA DE ACORDO COM DISTÂNCIA DE VISIBILIDADE DE PARADA	20
TABELA 5 - DISTÂNCIA DE SEGURANÇA DE ACORDO COM A "REGRAS DA METADE"	20
TABELA 6 - TESTES DE FRENAGEM REALIZADA PELA REVISTA AUTOMOTIVA [QUATRO RODAS, 2007]	22
TABELA 7 - ESPECIFICAÇÕES DAS LIGAÇÕES DO CIRCUITO DA FIGURA 9	33
TABELA 8 - PINOS DO DISPLAY DE LCD [SOUZA E LAVINIA, 2005]	36
TABELA 9 - AFERIÇÃO DA PRECISÃO DO SENSOR.....	61

Lista de Siglas

A/D –	Analógico/Digital
ASCII –	American Standard Code for Information Interchange
CGRAM –	Caracter Generator RAM
E/S –	Entrada e Saída
EEPROM -	Electrically-Erasable Programmable Read-Only Memory
LCD –	Liquid Cristal Display
LED –	Lighting Emissor Diode
RAM –	Random Access Memory
RISC -	Reduced Instruction Set Computer
ROM –	Read Only Memory
RS232 –	Recommended Standard 232
ULA -	Unidade Lógica e Aritmética

Capítulo 1 – Introdução

Atualmente os automóveis estão mais seguros, utilizando-se dos componentes eletrônicos que auxiliam o motorista a controlar melhor seu veículo e com isso reduzir os acidentes. Diversas empresas especializadas em dispositivos eletrônicos automotivos desenvolvem esses recursos de segurança, como por exemplo a nacional *TRW Automotive* e a alemã *Bosch®* [Bosch, 2008]. Muitos fabricantes de veículos já equipam seus automóveis com esses dispositivos, dentre as quais a sueca Volvo e a alemã Mercedes-Benz são tidas como referência em automóveis seguros.

Vários são esses recursos que melhoram a segurança do veículo e podem ser de dois tipos: passivo e ativo. Dispositivos de segurança passivos são aqueles que agem quando o acidente ocorre e servem para reduzir os danos dos ocupantes. Um exemplo de sistema passivo é o *air-bag*, que é uma bolsa inflável que reduz o impacto do ocupante do automóvel contra o volante. O sistema do tipo ativo é aquele que age antes do acidente ocorrer, tentando evitá-lo. O *Anti-blocking System* (ABS), Sistema Anti-bloqueio, é um exemplo de dispositivo de segurança ativa em que um componente eletrônico atua na frenagem do veículo, evitando que as rodas travem em uma frenagem brusca. Esse sistema é muito interessante em pisos com pouca aderência, pois além de não deixar que as rodas travem, faz com que o condutor tenha controle do veículo [Bosch, 2008].

O projeto desenvolvido nesta monografia é um dispositivo de segurança do tipo ativo, que alerta o motorista da proximidade do veículo que

segue a frente e com isso tenta alertar o condutor do veículo para que ele mantenha a distância de segurança frontal entre veículos no trânsito.

1.1 - Motivação

Atualmente o trânsito nas grandes cidades tem aumentado muito por vários fatores, por exemplo, a necessidade de percorrer longos trajetos e também pelo aumento de veículos, uma vez que a população conseguiu melhores condições de adquirir bens, sendo um deles o veículo automotivo. Um dos efeitos do aumento no trânsito é que nem todos os grandes centros comportam a quantidade de veículos que trafegam nas ruas, aumentando assim a quantidade de congestionamentos. Por consequência, isso gera um crescimento de riscos de acidentes de trânsito. [Kilsztajn et al., 2008].

Acidentes em vias urbanas dependem de diversas variáveis, tais como má condição da pista ou do veículo, desrespeito às normas de trânsito, distração dos motoristas entre outras. Estudos feitos pela *American Automobile Association Foundation for Traffic Safety*, e apresentados na **tabela 1**, indicam que 93% dos acidentes (dados de 1995 à 1999) são por erros humanos. Destes, 8,3% dos condutores estavam distraídos no momento do acidente. [AAA Foundation, 2001]

Tabela 1 - Tipos de distração e percentual de ocorrência [AAA Foundation, 2001]

Tipos de distração em trânsito (dados de 1995 a 1999)

Com objetos, pessoas ou evento fora do veículo	29,4%
Sintonizando rádio/usando CD Player	11,4%
Com outro ocupante no veículo	10,9%
Distração sem motivo aparente	8,6%
Movendo algum objeto no veículo	4,3%
Usando algum objeto	2,9%
Acionando ventilador ou ar condicionado	2,8%
Comendo ou bebendo	1,7%
Usando ou discando telefone celular	1,5%
Fumando	0,9%
Outros	25,6%

Outro fator causador de acidentes de trânsito é o desrespeito do condutor em guardar distância segura entre o automóvel que ele guia e o da frente indicada, na Lei 9.503 de 23 de setembro de 97, que trata do Código de Trânsito Brasileiro [LEI 9503/BRASIL, 2008]:

“Art. 23, inciso II,

O condutor deverá guardar distância de segurança lateral e frontal entre o seu e os demais veículos, bem como em relação ao bordo da pista, considerando-se, no momento, a velocidade e as condições do local, da circulação, do veículo e as condições climáticas;”

Esta lei indica que é necessário obedecer a distância de segurança dos carros que trafegam na via com a finalidade de tentar evitar acidentes, porém a distância de segurança é definida variando de acordo com diversos fatores.

Pensando em reduzir colisões frontais entre os veículos o estudo proposto nesse projeto final de graduação é do desenvolvimento de um protótipo usando componentes eletrônicos, incluindo um sensor de distância, que analisa em tempo real a velocidade atual do veículo e a distância que ele se encontra do carro que segue a frente. Com este dispositivo de segurança ativado o motorista é alertado se ele está afastado o suficiente do veículo da frente com a finalidade de evitar ou mesmo diminuir as conseqüências de uma eventual colisão frontal.

1.2 Objetivos

O objeto desta monografia é o desenvolvimento de um protótipo de segurança ativa para automóveis que alerta o motorista de que a distância entre o veículo dirigido e o que segue a frente deste é segura. Assim, o condutor terá tempo e espaço para reagir a um possível acidente ou pelo menos reduzir os danos se este ocorrer. São objetivos específicos deste projeto:

- Verificar a distância segura frontal entre os veículos para determinadas velocidades, a partir de pesquisa de referências disponíveis;
- Determinar a distância segura para implementação do protótipo;
- Montagem dos componentes eletrônicos;
- Desenvolver a estrutura lógica de controle do protótipo;
- Desenvolver um modelo em escala reduzida do ambiente real para demonstração do projeto.

1.3 Metodologia

O desenvolvimento deste projeto dar-se a partir de pesquisas bibliográficas para obtenção dos dados de distância de segurança, tais como manuais de direção defensiva e também livros de física e mecânica. Para o protótipo são utilizados:

- Um kit de desenvolvimento programável usando um microcontrolador da Microchip 16F877A que realiza:
 - Controle dos dados – Calcular distância segura, comparar com distância captada do sensor, converter o sinal analógico do sensor em sinal digital;
 - Simulação da velocidade atual do veículo.
- Um sensor de distância do tipo ultra-sônico modelo LV-MaxSonar® -EZ1™ da MaxBotix® Inc., para medição da distância.
- Um alarme sonoro tipo *buzzer*, que sinaliza distância insegura para o condutor, e um LED que indica distância de segura.

Para demonstração do protótipo é utilizado um modelo em escala reduzida, obedecendo às limitações do sensor de distância ultra-sônico e microcontrolador, bem como contribuindo para redução dos custos e levando em consideração a finalidade acadêmica do projeto. O ambiente a ser simulado é de veículos de passeio em bom estado de conservação (pneus e freios em boas condições), pista seca e condutor em estado normal.

1.4 - Estrutura da Monografia

Esta monografia é composta por 7 capítulos com os seguintes tópicos:

Cap. 1 – INTRODUÇÃO

Este capítulo contempla uma análise da abrangência do projeto, seus objetivos gerais e específicos, benefícios esperados e um panorama descritivo dos capítulos da monografia. Faz menção aos aspectos que motivaram o projeto com destaque para os objetivos.

Cap. 2 – ESTUDO DA DISTÂNCIA SEGURA

Neste capítulo é abordado um estudo sobre a distância segura, conceitos físicos e métodos usados nos cursos de direção defensiva. É decidido o método adotado para a implementação do projeto. Apresenta referências sobre estudos de acidente de trânsito e direção defensiva.

Cap. 3 – SENSOR DE DISTÂNCIA – CARACTERÍSTICAS, FUNCIONAMENTO, LIMITAÇÕES.

Neste capítulo são abordadas as características técnicas de funcionamento do sensor, suas limitações e o motivo da escolha do sensor de distância ultra-sônico.

Cap. 4 – KIT DE DESENVOLVIMENTO MULTIPIC – CARACTERÍSTICAS, FUNCIONAMENTO, DETALHES TÉCNICOS

Neste capítulo são abordadas as características técnicas, funcionamento e o motivo da escolha do kit com o microcontrolador PIC 16F877A.

Cap. 5 – IMPLEMENTAÇÃO

Neste capítulo são descritas as dificuldades apresentadas, métodos e ferramentas utilizadas.

Cap. 6 - TESTES E RESULTADOS

Neste capítulo é demonstrada a avaliação dos testes e a avaliação dos resultados.

Cap. 7 – CONCLUSÕES E TRABALHOS FUTUROS

Neste capítulo são descritas as principais conclusões obtidas e sugestões para trabalhos futuros.

Capítulo 2 – Definições de distância de segurança

frontal entre veículos

Neste capítulo é definida a distância segura que o condutor deve manter para que ele tenha tempo de reagir tentando evitar colisões com o veículo que trafega a frente. Os conceitos abordados neste capítulo são o tempo de reação do motorista em acionar o freio, o tempo para o veículo parar de acordo com testes realizados por revistas automotivas, conceitos de física que envolvem a distância segura e fundamentos de direção defensiva relacionados a distância de segurança adotado pelo Denatran e outras entidades de trânsito internacionais.

2.1 – Direção Defensiva - Distância segura

“Direção defensiva ou direção segura é a melhor maneira de dirigir, pois ajuda a preservar a vida, saúde e meio ambiente. Esse é o modo de dirigir que permite reconhecer antecipadamente situações de perigo e prever o que pode acontecer com seu veículo. Para isso, o condutor precisa aprender os conceitos de direção defensiva e usar esse conhecimento ao seu favor.”
[DENATRAN, 2005]

Muitas situações no trânsito acontecem automaticamente. Esse automatismo no trânsito esconde um problema que está na base de muitos acidentes, pois embora muito atento, em condições normais, o cérebro demora alguns tempo para reagir a determinadas situações. [DENATRAN, 2005]

Como exemplo, seja a seguinte situação:

A 80 km/h um veículo percorre 22,2m em apenas 1 segundo, o que significa percorrer em torno de 5 vezes o tamanho de um carro popular, que tem em média 4m. Se acontecer uma emergência, entre perceber o problema e tomar a decisão de frear, acionar o pedal de freio e por fim o veículo parar será necessário manter, pelo menos, 44,4 metros do obstáculo à frente. Essa distância é considerada para veículos em boas condições, freios convencionais (sem ABS) e pista seca. [DENATRAN, 2005]

A **figura 1** ilustra a situação descrita no exemplo. Isso demonstra que a esta velocidade um veículo pode sofrer sérios danos em um acidente, bem como seus ocupantes.

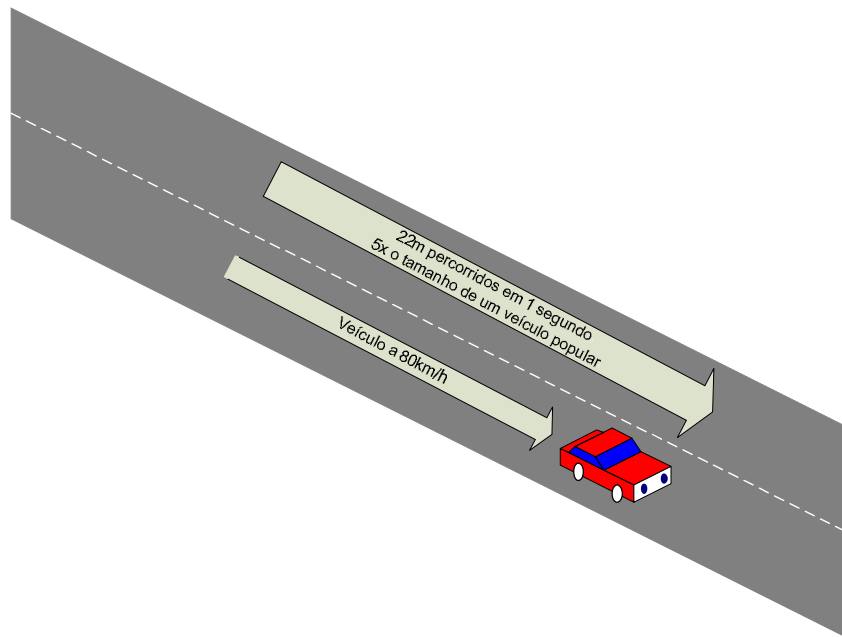


Figura 1 – Distância percorrida a 80 km/h



Figura 2 - Consequência de colisão de veículo depois de freada brusca (Fonte: G1.com)

Contudo manter a distância de segurança é fundamental para que o motorista aviste o veículo da frente freando e tenha tempo para reagir a uma situação de emergência, neste caso uma colisão evidente, seja desacelerando ou mesmo desviando.

A guarda da distância segura é relativa a diversas condições. São algumas:

- Estado de conservação do veículo tal como suas características (tipo de freio);
- Condições climáticas;
- Condições da via;
- Reação do condutor;
- Velocidade do veículo.

O Código de Trânsito Brasileiro também cita no CAPÍTULO III DAS NORMAS GERAIS DE CIRCULAÇÃO E CONDUTA, Art. 23, inciso II a guarda da distância segura entre os veículos, mas não detalha qual é a

distância ideal em função da variação das condições acima. [LEI 9503/BRASIL, 2008]

Desta forma, um dos métodos para verificar a distância segura, ensinado pelo Departamento Nacional de Trânsito (DENATRAN), é de manter o tempo de 2s do veículo à frente para carros com até 6m de comprimento, em pista seca e para velocidades de até 90km/h. Para pista molhada ou veículos maiores que 6m utiliza-se 3s ou 4s. A *The Royal Society for the Prevention of Accidents* [ROSPA, 2008] também adota essa regra chamada de “regra dos dois segundos”.

Para verificar o tempo de 2s do carro da frente é adotado o seguinte procedimento: [DENATRAN, 2005] [ROSPA, 2008]

- 1- Escolher um ponto de referência na pista. (poste, árvore, etc)
- 2- Logo que o veículo da frente ultrapassar esse ponto de referência o condutor deverá contar, pausadamente, cinqüenta e um, cinqüenta e dois, pois contando desta forma é o tempo aproximado de dois segundos.
- 3- Se o veículo do condutor tiver passado o ponto de referência antes de terminada a contagem isto indica que ele deve afastar-se do veículo da frente.
- 4- Caso contrário, terminada a contagem e o carro do condutor não tiver passado da referência, isto significa que o condutor está mantendo a distância segura mínima necessária.

Neste procedimento é possível que o motorista perca a concentração na via podendo provocar um acidente, o que reforça a utilização de um recurso eletrônico que o auxilie.

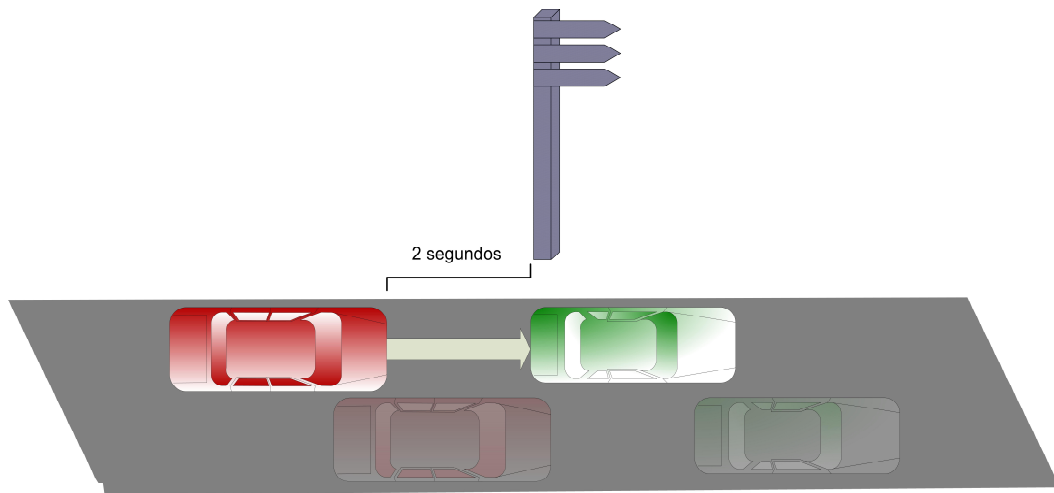


Figura 3 - Diagrama da regra dos dois segundos

Usando este método pode-se calcular a distância percorrida em determinadas velocidades usando a seguinte equação da cinemática:

$$D_s = V_1 . t \quad (2.1)$$

onde D_s é a distância segura, V_1 é a velocidade atual do veículo em m/s e t é o tempo. Na referida regra usa-se 2s.

Esta equação é derivada da função horária dos espaços do movimento retilíneo uniforme [Nussenzveig, 2002], onde um ponto material percorre espaços iguais em intervalos de tempos iguais. [Aragão, 2003]

Há outra regra empírica usada para calcular a distância segura, onde se divide a velocidade atual do veículo (em quilômetros por hora) por dois e o resultado será a distância de segurança em metros. [Dotta, 2000, pag. 102]

$$Ds = V/2 \quad (2.2)$$

Ou seja, para 100km/h deve-se manter 50m do carro da frente, para 80km/h serão 40m e 60km/h, 30m. Para indicação, esta regra será chamada de “regra da metade”. Esse método é de difícil verificação pois o motorista não tem referência da distância.

Essas duas regras são abstraídas de deduções práticas para que o condutor possa avaliar de alguma forma qual a distância segura que deve ser mantida.

2.2 – Dinâmica de frenagem e distância de visibilidade de parada

Esta seção aborda a dinâmica da frenagem e a distância de visibilidade de parada.

Como visto na **seção 2.1** a distância segura é determinada com base em diversos fatores como, por exemplo, velocidade do carro e condição do motorista (que varia seu tempo de reação a um possível acidente). São mostrados os aspectos físicos da frenagem considerando o tempo de reação do motorista e o tempo de frenagem.

2.2.1 - Conceitos físicos da frenagem

Um veículo em movimento uniforme percorre espaços iguais em tempos iguais e em velocidade constante diferente de zero. Esse movimento é chamado de Movimento Uniforme (MU) [Aragão, 2003] e é descrito pela seguinte função horária dos espaços:

$$S(t) = S_0 + V \cdot t \quad (2.3)$$

onde $S(t)$ é a posição em um determinado instante t , S_0 é o espaço inicial, V é a velocidade e t o tempo.

Em uma frenagem o veículo sofre desaceleração ou movimento retardado, que é o decréscimo da velocidade com o passar do tempo [Aragão, 2003]. A aceleração média do veículo é definida pela seguinte equação;

$$a = \frac{\Delta V}{\Delta t} = \frac{V - V_0}{t - t_0} \quad (2.4)$$

O movimento envolvido na frenagem é chamado de movimento uniformemente variado (MUV) [Aragão, 2003] [Nussenzveig, 2002], que neste caso é um movimento retardado. “O movimento será retardado,..., quando o valor absoluto da velocidade decresce; em cada unidade de tempo, o móvel percorre distância menor que na anterior” [Aragão, 2003, pág 168]. Para este movimento têm-se as seguintes equações:

Função horária dos espaços:

$$S(t) = S_0 + V_0 \cdot t + \left(\frac{1}{2} \cdot at^2\right) \quad (2.5)$$

Função horária da velocidade:

$$V(t) = V_0 + at \quad (2.6)$$

Equação de Torricelli:

$$V^2 = V_0^2 + 2a\Delta S \quad (2.7)$$

onde: $S(t)$ é a posição ocupada no tempo t , S_0 é a origem dos espaços, V_0 velocidade inicial (no tempo $t=0$), $V(t)$ é a velocidade no tempo t e a é a

aceleração ($a > 0$ se o movimento for acelerado; $a < 0$ se for retardado, para uma velocidade positiva). [Aragão, 2003]

Para este estudo consideram-se alguns princípios da dinâmica:

1ª Lei de Newton ou Princípio da Inércia: “Todo corpo persiste em seu estado de repouso, ou de movimento retilíneo uniforme, a menos que seja compelido a modificar esse estado pela ação de forças impressas sobre ele.” [Nussenzveig, 2002]

2ª Lei de Newton (Princípio Fundamental da Dinâmica): “A resultante das forças que atuam em um ponto é igual ao produto de sua massa pela aceleração adquirida.” [Aragão, 2003] [Nussenzveig, 2002]

$$F_R = m \cdot a \quad (2.8)$$

3ª Lei de Newton ou Princípio da ação-reação: “A toda ação corresponde uma reação de mesma direção e de mesma intensidade, em sentido contrário” [Aragão, 2003]

Esses princípios e equações são aplicados para o cálculo da distância de visibilidade de parada.

2.2.2 - Distância de visibilidade de parada

Para que o motorista tenha tempo de reagir tentando evitar um acidente ele deve manter a distância de segurança ou distância de visibilidade de parada, que é a distância mínima necessária para que um veículo que percorre em uma estrada possa parar antes de atingir um obstáculo na sua trajetória. [Negrini Neto, 2003]

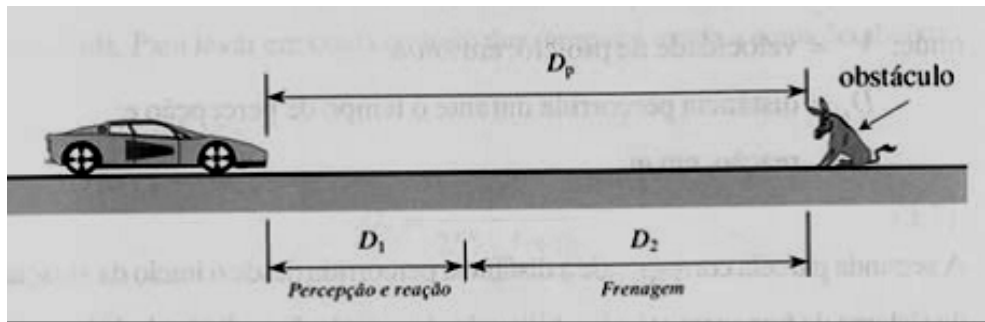


Figura 4 - Distância de visibilidade de parada (Fonte: Topografia Geral)

Na **figura 4** tem-se que a distância de visibilidade de parada (D_p), que será chamada de distância segura, é composta por duas distâncias D_1 e D_2 , que são respectivamente a distância de percepção e reação e distância de frenagem.

O tempo de percepção é o período que o motorista demora a perceber um obstáculo diante de seu veículo. O tempo de reação é o tempo que o motorista demora para acionar o freio do veículo tentando evitar um acidente. Esse tempo de reação e percepção é diretamente ligado à condição do motorista, e é afetado caso o motorista esteja sob efeitos de medicamentos ou algum tipo de droga, de acordo com a iluminação da via ou idade do condutor, entre outros. [Aragão, 2003]

Um estudo sobre tempo de reação apontado por Araújo indica que o tempo percepção-resposta (percepção+reação), estando o motorista em condições normais, apresenta uma variação de 0,75s a 1,5s. Neste estudo o tempo de reação médio escolhido para todas as deduções práticas foi de 1s. [Aragão, 2003, pág. 234]

No sítio [Visualexpert, 2008] existe um resumo de um estudo chamado “*How Long Does It Take To Stop?*”. Este resumo descreve sobre o tempo de reação do motorista em acionar o freio. Nele foi indicada uma média de 1,5s para o motorista comum acionar o freio.

No projeto desta monografia é tomado o tempo médio de 1s seguindo a aplicação feita por [Aragão, 2003] em seus estudos.

Com esses dados pôde-se verificar a distância percorrida em movimento uniforme em D_1 , usando a seguinte equação da cinemática derivada de (2.3):

$$D_1 = V \cdot T_R \quad (2.9)$$

onde, T_R é o tempo de reação ($T_R=1s$) e V é a velocidade atual do veículo, lembrando que deve ser convertida de km/h para m/s. Desta forma, segue o seguinte exemplo para a velocidade de 80km/h:

$$80km/h = 22,2m/s$$

$$D_1 = 22,2 \cdot 1s = 22,2m$$

Para o cálculo da distância percorrida durante a frenagem D_2 do veículo usa-se a equação a seguir:

$$D_2 = \frac{(V_0)^2}{2 \mu g} \quad (2.10)$$

Esta equação é derivada da equação de Torricelli do Movimento Uniformemente Variado (MUV), (2.7) considerando $V = 0$ e a aceleração (a),

negativa, igual ao produto da aceleração da gravidade ($g=9,8m/s^2$) pelo coeficiente de atrito (μ) [Nussenzveig, 2002].

$$a = -\mu g \quad (2.11)$$

Para o coeficiente de atrito segue a **tabela 2** para determinadas condições da via:

Tabela 2 - Coeficiente de atrito em alguns tipos de pavimento [Aragão, 2003]

Pavimento	Seco	Molhado
Asfalto	0,80	0,70
Concreto	0,70	0,40
Paralelepípedo	0,70	0,50
Pedra / areia	0,55	0,40

Calculando a distância percorrida durante a frenagem (D_2) para o veículo à 80km/h no asfalto seco:

$$80km/h = 22,2m/s$$

$$D_2 = [(22,2)^2 / (2 \cdot 0,8 \cdot 9,8)] = 31,4m$$

Neste caso são necessários 31,4m para o veículo, nas condições descritas acima, parar por completo.

Calculados D_1 e D_2 obtém-se a distância de parada (D_P):

$$D_P = D_1 + D_2 \quad (2.12)$$

$$22,2m + 31,4m = 53,6m$$

Deste modo para um veículo a 80 km/h, considerando o coeficiente de atrito em asfalto seco ($\mu=0,8$), pista plana (sem declives ou aclives) e tempo

de reação do motorista de 1s, a distância de visibilidade de parada mínima que deve ser mantida pelo condutor é de 53,6m.

2.3 - Estudo comparativo entre os métodos de cálculo de distância segura

Nesta seção é feito um estudo comparativo entre a equação de distância de visibilidade de parada, as regras descritas na **seção 2.1** e as distâncias reais de frenagem de alguns veículos com base em dados de revistas automotivas.

Primeiramente, utilizando a regra dos dois segundos tem-se que a distância segura é manter 2s do automóvel que segue a frente para velocidade de até 90km/h, segundo manual de direção defensiva do DENATRAN [DENATRAN, 2005]. Deste modo segue a **tabela 3** com o resultado desse método:

Tabela 3 - Distância de segurança usando a "regra dos dois segundos"

Velocidade atual do veículo	Distância segura a ser mantida pela regra dos 2 segundos
40km/h (11,1m/s)	22,2m
60km/h (16,7m/s)	33,4m
80km/h (22,2m/s)	44,4m

Para calcular a distância de visibilidade de parada são usadas as equações **2.9**, **2.10** e **2.12**; Usando os mesmos valores de velocidade para o exemplo anterior e tempo de reação do motorista ($T_R = 1s$), coeficiente de atrito

($\mu = 0,8$) e aceleração da gravidade ($g = 9,81\text{m/s}^2$) os resultados são apresentados na **tabela 4** a seguir:

Tabela 4 - Distância de segurança de acordo com distância de visibilidade de parada

Velocidade atual do veículo	D_1	D_2	Distância segura a ser mantida pela distância de visibilidade de parada ($D_1 + D_2$)
40km/h (11,1m/s)	11,1m	7,8m	18,8m
60km/h (16,7m/s)	16,7m	17,7m	34,4m
80km/h (22,2m/s)	22,2m	31,4m	53,6m

Para a **tabela 5** foi usada a “Regra da metade” de acordo com **(2.2)**.

Tabela 5 - Distância de segurança de acordo com a "regra da metade"

Velocidade atual do veículo	Distância segura a ser mantida pela regra da metade ($D_S = V_{atual} / 2 \Rightarrow D_S$ em metros)
40km/h (11,1m/s)	20m
60km/h (16,7m/s)	30m
80km/h (22,2m/s)	40m

Comparativo entre os métodos de distância segura

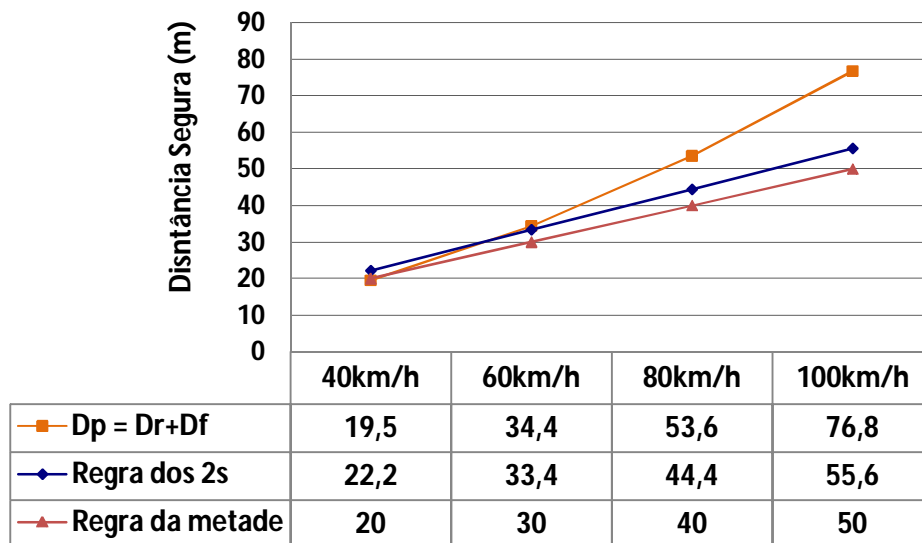


Gráfico 1 - Comparativo entre as distâncias de segurança

A regra da metade e a regra dos dois segundos apresentam linearidade para distância segura que podem ser vistas no **gráfico 1**. Essas duas regras, embora usadas em cursos de direção defensiva, até pela facilidade de verificação, não apresentam muita precisão por considerar apenas a velocidade do veículo, mas apresentam proximidade do valor da regra distância de visibilidade de parada (D_p). Também é possível ver que elas coincidem por pequena diferença à velocidade de 40 km/h. Analisando a distância segura para 80km/h tem-se uma diferença de até 13,6m entre os métodos aumentando de acordo com o aumento da velocidade. Entretanto, as duas regras práticas, da metade e dos dois segundos, são válidas para o trânsito em áreas urbanas onde a velocidade dificilmente supera os 80km/h e onde há maior tráfego de automóveis.

Para maior precisão nos cálculos é utilizado o método Dp que apresenta outras variáveis que influenciam na frenagem do veículo e desta forma pode ser utilizado em melhorias futuras.

Na **tabela 6** são citados os dados de alguns veículos com frenagens de 60km/h à 0km/h e de 80km/h à 0km/h, testes realizados por revistas automotivas em condições reais com veículos de freio convencional (sem ABS). Neste caso constata-se que a equação dada por Torricelli coincide com alguma margem de erro da velocidade calculada. A margem de erro é calculada com base na distância de frenagem da **equação 2.10**.

Tabela 6 - Testes de frenagem realizada pela revista automotiva [Quatro Rodas, 2007]

Veículo	Frenagem de 60km/h – 0 (margem de erro)	Frenagem de 80km/h – 0 (margem de erro)
Renault Logan 1.6 8V	19,9m (12,4%)	33,5m (6,6%)
Ford Fiesta Sedan 1.6 8V	15,2m (14,1%)	28,9m (7,9%)
Chevrolet Prisma 1.4 8V	18,4m (3,9%)	31,1m (0,9%)
Fiat Siena 1.8 8V	19,8m (11,8%)	30,4m (3,1%)
Corsa Hatch 1.4V	16,6m (6,2%)	27,8m (11,4%)

Conclui-se neste estudo que o método mais preciso de verificação de distância segura é o cálculo da distância de visibilidade de parada, pois este inclui mais parâmetros que interferem na distância de segurança, tais como coeficiente de atrito, velocidade e tempo de reação do motorista. Desta forma o protótipo pode receber melhorias futuras com a alteração dos valores mencionados para admitir o melhor resultado para determinada situação. Além disso, este método iguala ou supera a margem de segurança das outras regras em velocidades superiores como pode ser visto no **gráfico 1**.

Capítulo 3 – Sensor de distância ultra-sônico MaxBotix

EZ1™

Sensores são elementos provedores de informações para os sistemas de automação industrial, residencial, automotiva e demais setores. Um sensor pode ser definido como um transdutor que altera a sua característica física interna devido a um fenômeno físico externo [Rosário, 2005]. São recursos bastante utilizados atualmente no meio industrial, veicular, residencial entre outras áreas, permitindo que dispositivos eletrônicos tenham uma maior interação com o meio externo. Segundo Rosário, “Um sensor muda seu comportamento sob a ação de uma grandeza física, podendo fornecer direta ou indiretamente um sinal que indica essa grandeza e convertendo uma quantidade física em um sinal elétrico.” [Rosário, 2005]

Dentre os sensores, destaca-se o sensor de proximidade nos seus variados tipos: os magnéticos, o óptico e o ultra-sônico.

Os sensores magnéticos são feitos em material ferro-magnético (ferro, níquel etc) e são acionados quando há uma variação do campo magnético. São largamente utilizados em sistemas onde é necessário identificar fim de curso e são classificados como capacitivos ou indutivos. Os indutivos identificam a alteração do campo magnético de materiais metálicos enquanto os capacitivos identificam a alteração da capacitância campo elétrico em objetos metálicos e não-metálicos. Este tipo de sensor não se aplica a este projeto, pois a distância identificada por ele é pequena, na faixa de alguns milímetros. [Rosário, 2005]

Os sensores ópticos são sensores que possuem um emissor e um receptor de luz, sendo o emissor um LED ou lâmpada e o receptor um componente fotossensível, como por exemplo, um fotodiodo. O funcionamento básico do sensor é baseado em um circuito oscilador que gera uma onda convertida em luz pelo emissor. Um objeto qualquer reflete a luz do emissor para o receptor e assim um circuito identifica a variação da luz recebida. Este sensor poderia ser utilizado neste projeto porém o sensor ultra-sônico utilizado atinge melhores condições de medição, como, por exemplo, maior alcance, mantendo custo semelhante. [Rosário, 2005]

Destes, o sensor ultra-sônico foi o escolhido para este projeto. A princípio a escolha deste sensor foi devido ao seu baixo custo de aquisição, R\$99,00 (ver **Apêndice 3**), fácil aplicabilidade (devido às diferentes formas de saída, três no total) e também pelo seu raio de ação de até 6m que serve ao modelo em escala reduzida.

3.1 – Princípio de funcionamento do sensor ultra-sônico

O princípio de funcionamento de um sensor ultra-sônico é semelhante ao dos sonares usados em embarcações. Estes sistemas funcionam da seguinte forma: Um sinal ultra-sônico é emitido em determinada direção. Este sinal sonoro rebate em algum obstáculo sendo refletido e captado pelo sensor. Deste modo, calcula-se o tempo gasto entre o envio do sinal e recebimento do eco. A **figura 5** ilustra esse funcionamento. [Rosário, 2005]

Ele funciona com o envio de um som com frequência acima de 20kHz, o ultra-som, inaudível ao ser humano (que escuta frequências de até 20kHz).

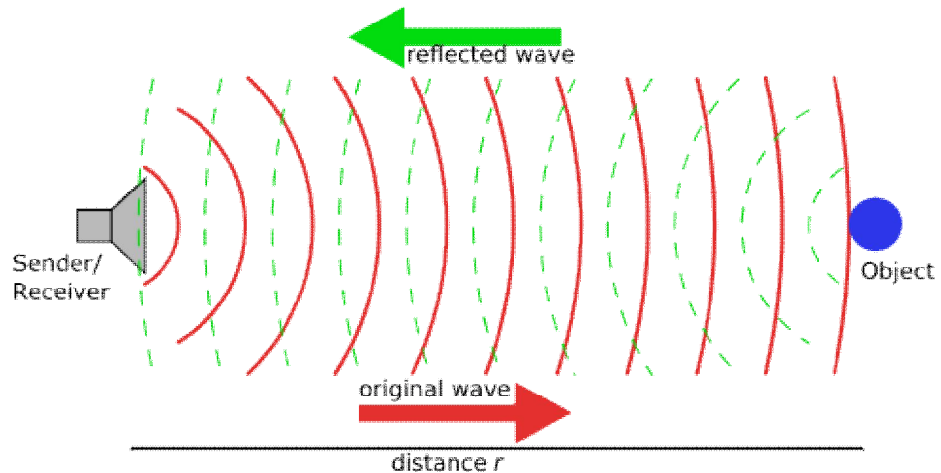


Figura 5 – Transmissão e recepção do eco ultra-sônico

Algumas das características desses sensores são: [MaxSonar, 2008]

- A utilização como sensores de proximidade;
- Frequência de trabalho entre 40kHz e 2MHz, podendo fornecer grande faixa de informações
- Podem apresentar problemas em ambientes com muito ruído (interferência)

O funcionamento do sensor ultra-sônico é baseado no efeito piezo-elétrico.

Rosário explica o efeito *piezoelétrico* da seguinte forma:

“...quando aplicamos uma força de tração, ou compressão no cristal, aparece uma tensão proporcional à força aplicada (conversão de força em tensão). E, quando se aplica uma tensão no cristal, ele se comprime ou expande automaticamente. Em decorrência, se aplicarmos uma tensão alternada em um cristal ele vibrará na mesma frequência da tensão aplicada (conversão de tensão em movimento).” [Rosário, 2005, pág. 66]

Desta forma, como apresentado na **figura 5**, o sensor ultra-sônico aplica uma tensão alternada em alta frequência no cristal, fazendo com que este vibre e emita um ultra-som. Este ultra-som é refletido em algum anteparo. Logo em seguida o cristal recebe o sinal ultra-sônico refletido que vibrando na mesma frequência converte esta vibração em tensão alternada. E, por fim, é calculado o tempo entre a emissão e o recebimento das ondas identificando a distância. [Rosário, 2005]

3.2 - Sensor ultra-sônico Maxbotix® MaxSonar® - EZ1™ – Características¹ e motivação de uso

3.2.1 – Características gerais

O sensor LV-MaxSonar – EZ1™ é um sensor ultra-sônico de alto desempenho para medições de distância mais estável que outros modelos de seus concorrentes. Dentre suas características estão seu alcance de medição que varia de 6” (0,15m) a 254”(6,35m). A resolução de leitura é de 1 polegada na saída serial digital. [Max Sonar, 2008]

Algumas características e benefícios do sensor são [Max Sonar, 2008]:

- Auto calibração – O sensor ao ser ligado faz uma calibração no primeiro ciclo.
- Baixo consumo de operação (na faixa de 2mA).
- Alimentação de 2,5V a 5,5V.
- Modos de operação automática ou manual utilizando o pino BW.
- 3 tipos de saídas de dados: PWM, Serial Digital e tensão analógica.

¹ Características retiradas do manual LV-MaxSonar ®-EZ1 High Performance Sonar Range Finder [Max Sonar, 2008]

- Frequência de operação de 42khz
- Baixo custo

3.2.2 – Motivação de uso do sensor Maxbotix EZ1

O projeto desta monografia é desenvolver um circuito eletrônico que monitora a distância segura que o condutor deve manter do veículo que trafega à sua frente. Para demonstração é montado um modelo em escala reduzida que represente a distância real, pois assim é possível reduzir os custos do projeto adquirindo um sensor que o autor pudesse arcar (até R\$300,00). Portanto, os pontos considerados para a escolha foram: viabilidade financeira para apresentação de fins acadêmicos, alcance mínimo de leitura de 3m (para que o modelo em escala não tivesse proporções muito reduzidas), velocidade de leitura de pelo menos de 100ms.

Depois de pesquisar alguns tipos de sensores de proximidade existentes no mercado, concluiu-se que o sensor que melhor se aplicava ao estudo era o que trabalhava com ultra-som. Na verificação dos preços constatou-se variações muito grandes, desde um modelo semelhante ao MaxSonar EZ1™ por R\$200,00 e o Sonar SRF04, disponível na [TATO, 2008] que custava R\$50,00, entre outros modelos de uso industrial, como exemplo o da Sick® de saída analógica e alcance de 6m que custava R\$2500,00. [SICK, 2009]

O modelo Sonar SRF04 tem um alcance de apenas 1,5m o que poderia tornar modelo de apresentação em escala muito reduzida. Além disso o Max Sonar tem maior estabilidade da leitura dos dados, segundo manual da MaxBotix, em comparação o SRF04. E, também, o sensor EZ1™ tem alcance

de até 6m de leitura. O sensor da Sick® tem custo muito elevado para um projeto de fim acadêmico. Sendo assim, o MaxBotix custou R\$99,00, tendo características que servem a este projeto e boa velocidade de leitura de 50ms, que equivale a 20 leituras por segundo.

Existem outros modelos diversos dos tipos infravermelhos, laser, capacitivo, indutivo e outros. Para a implementação de um produto comercial baseado neste protótipo em um automóvel seria interessante o uso do sensor tipo Radar que atinge frequências mais altas de operação como o modelo usado em carros da Mercedes-Benz e Volvo. [DISTRONIC, 2008] O modelo laser também tem longo alcance, porém para que a aferição da distância seja feita corretamente o feixe laser deve estar alinhado perpendicularmente com o receptor. Já os modelos capacitivos e indutivos não têm um longo alcance, em média alguns milímetros. Para estes não foram avaliados os preços.

A saída analógica foi escolhida para este projeto pois com o conversor A/D contido no PIC 16F877A é possível obter maior resolução e precisão da medição com 1024 níveis ante a serial digital do próprio sensor com 256.

3.2.3 - Dimensões e descrição das saídas de dados do LV - MaxSonar - EZ1™

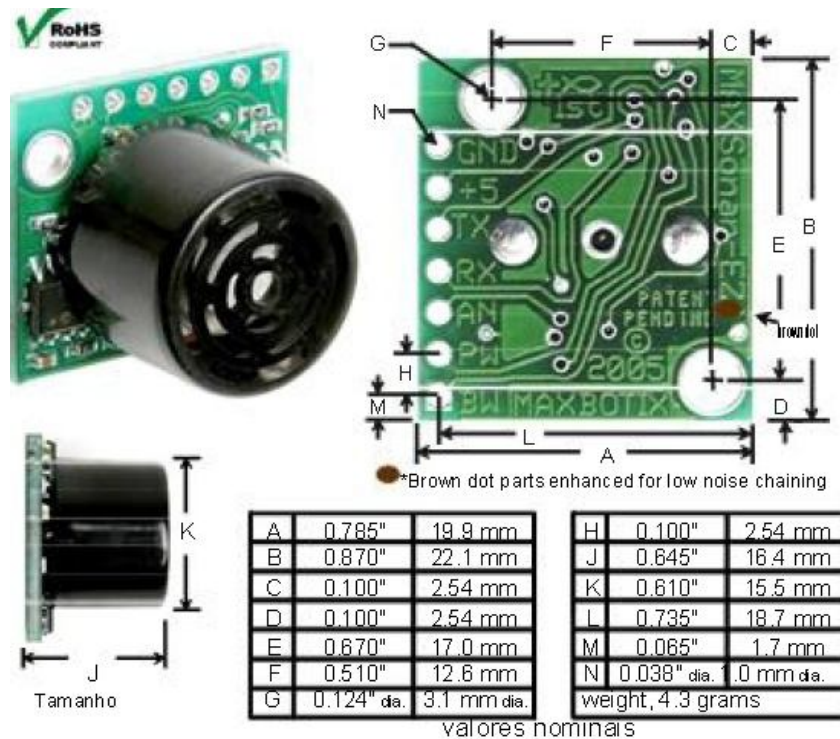


Figura 6 - Sensor MaxSensor EZ1™ Dimensões (Fonte: Max Sonar, 2008)

A **figura 6** ilustra as dimensões do sensor, destacando seu tamanho de 20x22mm. Seu tamanho reduzido facilita a montagem no protótipo.

Este sensor possui 3 tipos de saídas como citado em **3.2.1** que são:

- **Serial digital:**

Os dados seriais são transmitidos pelos pinos TX e RX. O pino TX envia os sinais seriais no padrão RS232 em modo assíncrono. A saída de dados é no formato ASCII "R", seguido por 3 dígitos ASCII que representam a distância em polegadas para um máximo de 256 polegadas seguido de um ENTER (código 13 da tabela ASCII). A velocidade de transmissão é de 9600bps, 8bits,

sem paridade, com um bit de parada. O pino RX é mantido em nível alto internamente (1 lógico). Isto significa que a medição é feita. Sinalizando o pino para nível baixo (0 lógico) a medição é cessada. [Max Sonar, 2008]

- **Analógica:**

Para a saída de tensão analógica é usado o pino AN que transmite 9,8mV/pol quando ligado em 5V e para 3,3V a tensão de saída é de 6,4mV/pol. Este é o pino utilizado para este projeto. A saída analógica atende os requisitos para demonstração em escala reduzida sendo de relativa facilidade na implementação, uma vez que o PIC utilizado no projeto possui o conversor analógico/digital, tem boa velocidade de conversão e maior resolução em comparação com o serial digital . [Max Sonar, 2008]

- **PWM – Modulação por Largura de Pulso:**

O pino PW envia sinais em modulação por largura de pulso. O fator de calculo é de 147 μ s por polegada ou 57,8 μ s por centímetro. [Max Sonar, 2008]

Os pinos +5V e GND são respectivamente os pinos de alimentação e aterramento. O pino BW é um pino de controle da saída dos dados que permite cessar, medir continuamente ou fazer uma única medição. No diagrama a seguir é possível verificar a posição dos pinos e composição do sensor. Os pinos e maiores detalhes do sensor podem ser identificados no diagrama que segue na **figura 7**. [Max Sonar, 2008]

Capítulo 4 – Kit de desenvolvimento MultiPIC

Neste capítulo são descritas as características do kit de desenvolvimento e do microcontrolador com ênfase naquelas que são utilizadas neste projeto.

4.1 – Descrição do Kit MultiPIC

Para o desenvolvimento deste projeto foi proposto a utilização de um kit que continha o microcontrolador *Intel 8051* disponível no laboratório de microcontroladores do Centro Universitário Uniceub. Este kit atendia os requisitos e era esperado usá-lo com a finalidade de minimizar os custos envolvidos neste projeto. Ao final de algum tempo foi verificado que adquirir um kit seria melhor para fazer testes em outros locais além do laboratório. O kit adquirido também apresenta compatibilidade com a linguagem C, que permite boas condições de inserir equações e tratamento de tipos de variáveis, possui LEDs soldados, sendo desnecessário montar esses em local separado. Nele também são oferecidos *trimpots*, resistores variáveis, que serviram para testes do conversor A/C sem a necessidade de ligar o sensor.

O kit escolhido foi o MultiPIC da SmartRadio [SmartRadio, 2008]. Este kit é baseado no microcontrolador PIC modelo 16F628. Essa ferramenta permite o uso de outros modelos PIC entre os quais é utilizado neste projeto o 16F877A que tem mais portas de entrada e saída de dados que o 16F628 e mais memória de programa, *8192kwords* contra *2048kwords*. Nele estão disponíveis (quando se usa o microcontrolador 16F877A) [16F877XA, 2009]:

- 2 chaves tácteis – Utilizado neste projeto para aumentar ou decrementar a velocidade simulada do carro-modelo;
- 4 *LEDS* vermelhos – Apenas 1 é utilizado neste projeto para sinalizar condição de distância segura
- Display de *LCD* de 16X2 – Informar a distância que o veículo mantém com o da frente, velocidade e distância de segurança para a velocidade atual.
- *Trim pots* - para uso com conversor A/D e testes da conversão sem utilizar o sensor
- 2 *displays* de 7 segmentos – não utilizados neste projeto e portanto desativado
- Conversor RS232/TTL – não utilizado e desativado
- Programador incorporado – Para gravar diretamente o código no PIC dispensando retirá-lo do kit para esse procedimento.
- Fonte de alimentação de +5V independente que dispensa alimentação adicional para ligar outros periféricos.

Na **tabela 7** é descrito onde estão conectados os principais dispositivos do kit com suas referências do PIC e do MULTIPIC.

Tabela 7 - Especificações das ligações do circuito da figura 9 [SmartRadio, 2008]

<i>Pinos do PIC</i>	<i>Referência no MultiPIC</i>	<i>Periféricos disponíveis</i>
A0, A1, A2 e A3	PA0, PA1, PA2 e PA3	LEDs
A4 e A5	PA4 e PA5	Chaves tácteis
B0, B1, B2, B3, B4 e B5	PB0, PB1, PB2, PB3, PB4 e PB5	Display de LCD – Pinos de dados e
C7 e C8	PC7 e PC8	Conversor TTL/RS 232
A0 e A1	PA0 e PA1	Trim pots (RV1 e RV2)

Os componentes do kit podem ser vistos na **figura 8**.

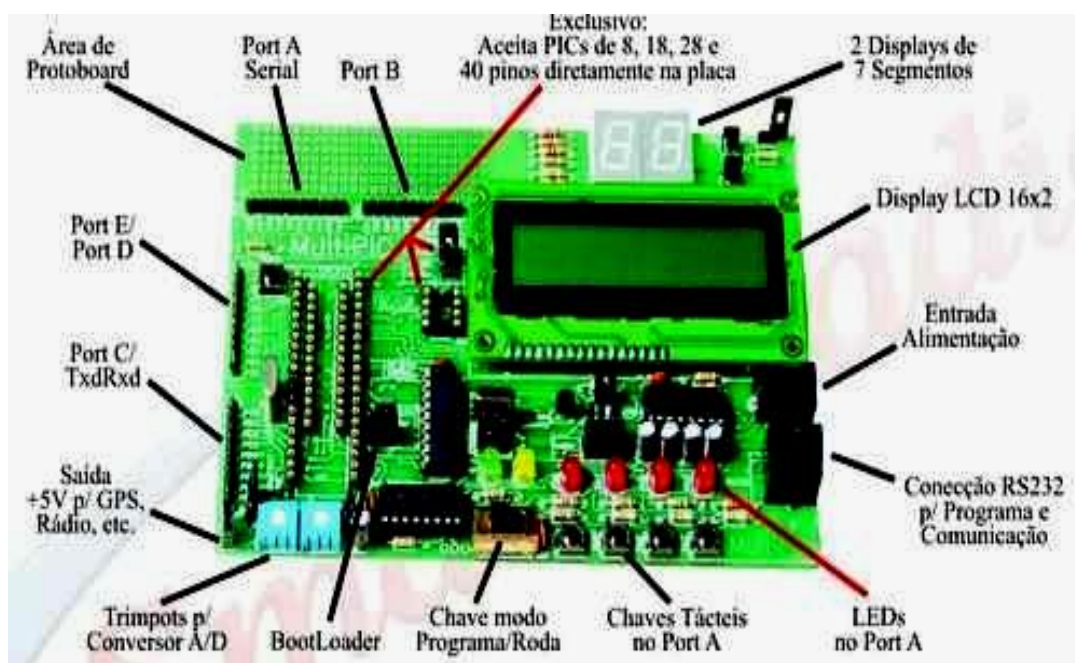


Figura 8 - Kit de desenvolvimento MultiPIC (Fonte: SmartRadio, 2008)

Este kit contém *jumpers* que servem para habilitar ou desabilitar seus periféricos. Utilizando um Pic de 40 ou 28 pinos pode ser utilizado um oscilador externo de 4MHz ligado aos pinos 13 e 14, que são OSC IN e OSC OUT respectivamente. Esse oscilador é mais preciso que o oscilador interno do

PIC e serve para maior precisão de *clock* quando se utiliza a comunicação serial RS232 em modo de operação síncrona.

Há uma chave de modos Programa e Roda. Essa chave serve para gravar o código no PIC (posição “Programa”) e para execução do algoritmo (posição “Roda”). A área de *protoboard* serve para fixação de periféricos externos como, por exemplo, o *buzzer* a ser utilizado neste projeto.

No diagrama da **figura 9** pode-se ver as ligações do circuito e respectivos periféricos do MultiPIC.

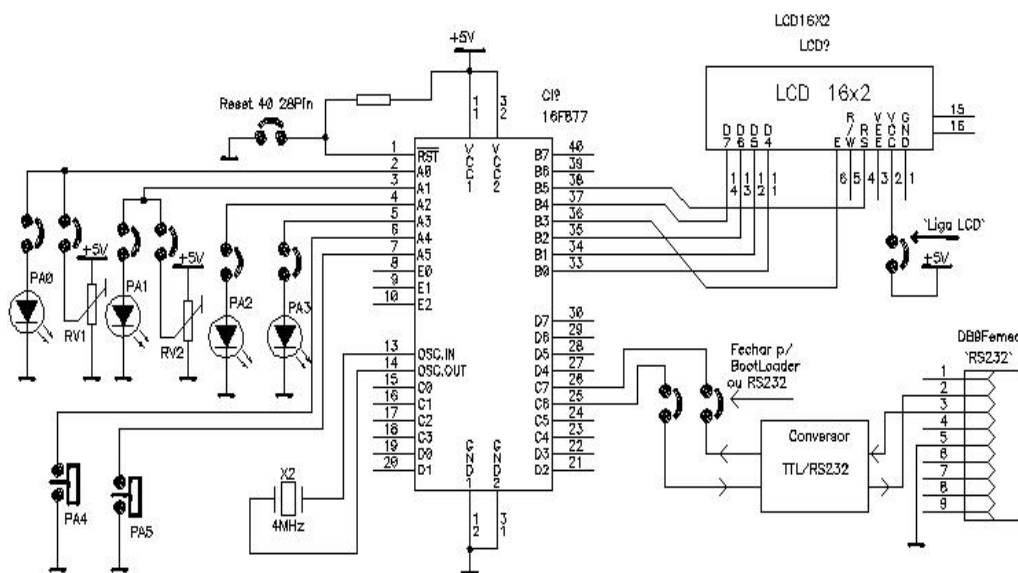


Figura 9 - Diagrama do circuito do Multipic (Fonte: SmartRadio, 2008)

A razão da escolha deste kit foi de facilitar a montagem do protótipo, sendo desnecessário montar os componentes separadamente em placa separada. Isso melhora a confiabilidade evitando maus-contatos e outros problemas dessa natureza. Este kit também já tem programador integrado que permite gravar o programa sem necessidade de retirar o microcontrolador.

4.1.1 – Display de LCD

O *display* de LCD utilizado no MultiPIC é um modelo de 16X2 caracteres da HANTRONIX. Esse *display* tem iluminação própria facilitando a visualização da escrita. Ele possui um controlador interno com 80bytes de memória RAM, 64 bytes de RAM para gerador de caracteres do usuário (CGRAM) e 9920 bits de memória ROM. [Pereira, 2005]

O *display* possui 16 pinos como segue a **tabela 9**:

Tabela 8 - Pinos do display de LCD [Souza e Lavinia, 2005]

1	V _{SS}	9	DB2
2	V _{DD}	10	DB3
3	V ₀	11	DB4
4	RS	12	DB5
5	R/W	13	DB6
6	E	14	DB7
7	DB0	15	A
8	DB1	16	K

Os pinos V_{SS} e V_{DD} são os de alimentação e variam entre 4,75V e 5,25V.

O pino **RS** (*Register Select*) serve de controle do tipo de informação passada para o *display*, utilizando 0 para um comando ou 1 para um dado.

O pino **R/W** serve para alterar o estado do visor entre Leitura/Escrita (Read/Write). Serve para habilitar a escrita de um dado no LCD ou para verificar se foi escrito o dado correto.

Os pinos **DB0** e **DB7** são os pinos para dados. No caso do MultiPIC apenas 4 pinos são ligados. Os pinos **A** (Anode) e **K** (Katode) servem para ligar a iluminação do display. [Souza e Lavinia, 2005]

A utilização deste *display* no kit é feita com a comunicação no modo de 4bits com a linha mais significativa, nos *ports* D4 ao B0, D5 ao B1, D6 ao B2 e D7 ao B4, podendo ser vista na **figura 9**. Desta forma, dividindo o byte em dois *nibbles* a transferência é feita iniciando pelo mais significativo.

4.2 –Características do Microcontrolador PIC 16F877A

Os Microcontroladores são componentes eletrônicos semelhantes aos microprocessadores que contém processador, memória e funções. Permitem controlar entrada e saída de informações e são muito utilizados por seu baixo custo e consumo de energia. [Souza e Lavinia, 2005]

As principais diferenças entre um microcontrolador e um microprocessador usado em computadores pessoais são a capacidade de armazenamento de memória, frequência de operação e quantidades de funções. Também é diferente a quantidade de recursos em uma só pastilha, como a quantidade de portas de entrada/saída, conversores analógico-digitais e PWM. [Souza e Lavinia, 2005]

O microcontrolador PIC é fabricado pela *Microchip Technology*. Dentre os modelos disponíveis está o PIC 16F877A. Estes microcontroladores

têm a arquitetura RISC significando que a quantidade de instruções que um microcontrolador contém é menor. Entretanto essa arquitetura é mais rápida servindo para fins mais específicos. [Souza e Lavinia, 2005]

A estrutura de máquina interna dos microcontroladores PIC é do tipo Harvard, onde existem dois barramentos internos: um de dados e outro de instruções. Para o PIC o barramento de dados é de 8bits e o de instrução pode ser de 12, 14 ou 16bits. Essa característica confere-lhe maior velocidade, pois, assim, uma instrução pode ser executada enquanto a outra é resgatada na memória. O PIC 16F877A, utilizado neste projeto, utiliza 14bits de instrução. [Souza e Lavinia, 2005]

As características básicas que o compõe são: [16F877XA, 2008]
[Souza e Lavinia, 2005]

- **5 conjuntos de portas de E/S totalizando 33 portas;**
 - PORTA, que é composta de E/S digitais bidirecionais e sistema analógico e é composta dos pinos:
 - RA0/AN0, RA1/AN1, RA2/AN2/V_{ref}-/CV_{ref}, RA3/AN3, RA4/T0CKI/C1OUT, RA5/SS/AN4/C2OUT.
 - PORTB, que é composta de E/S digitais bidirecionais e pode ser ligado/desligado via *software*. 5 pinos estão associados às interrupções. É composta dos pinos:
 - RB0/INT, RB1, RB2, RB3/PGM, RB4, RB5, RB6/PGC, RB7/PGD
 - PORTC, também é composta de E/S digitais bidirecionais, com os pinos:

- RC0/T1OSO/T1CKI, RC1/T1OSI/CCP, RC2/CCP1, RC3/SCK/SCL, RC4/SDI/DAS, RC5/SDO, RC6/TX/CK, RC7/RX/DT.
- PORTD, que é composta por E/S bidirecionais ou porta de comunicação paralela composta pelos pinos:
 - RD0/PSP0, RD1/PSP1, RD2/PSP2, RD3/PSP3, RD4/PSP4, RD5/PSP5, RD6/PSP6, RD7/PSP7.
- PORTE, que é composta por E/S bidirecionais e sistema analógico composto pelos pinos:
 - RE0/RD/AN5, RE1/WR/AN6, RE2/CS/AN7.
- **Memória RAM;**
 - Esta é a memória de dados do sistema que é usada para guardar as variáveis e registradores utilizados pelo programa enquanto esse é executado. Nesta memória são armazenados dados de 8bits e ela é volátil, isto é, quando o PIC é desligado os dados contidos nela são perdidos.

Essa memória está dividida em dois grupos: registradores especiais e registradores de uso geral. Os primeiros são utilizados pelo microcontrolador para processar a ULA e também para o programa. Servem para configuração de várias funções e periféricos, podendo ser escritas tanto pelo usuário quanto pelo *hardware*. O segundo registrador é destinado para guardar variáveis que o usuário define para utilização no programa. Para isso, são destinados 368bytes dessa memória.

- **Memória *flash* de 8kwords;**

- A memória Flash é onde fica armazenado o código do programa. Esta pode ser gravada mais de 100mil vezes no caso do 16F877A.
- **Memória EEPROM de 256bytes;**
 - Esta é uma terceira memória do PIC para armazenamento de dados. Também é uma memória não-volátil que armazena os dados mesmo sem alimentação.
- **Frequência de 20Mhz de operação;**
 - Este modelo de PIC tem oscilador interno que permite frequências de até 20Mhz selecionável via linha de comando. Para maior precisão dos dados é preferível usar um oscilador externo normalmente do tipo cristal. O kit MultiPIC conta com um oscilador externo de até 4Mhz. Para utilização do oscilador externo pelo menos 1 pino será inutilizado para outros fins.
- **35 instruções básicas.**
 - São 4 grupos principais de instruções em *assembly*: Transferência de Dados, Lógica e Aritmética, Operações sobre bits e direcionamento do programa.

4.2.1 – Interrupções do PIC 16F877A

“A interrupção é dada como um evento externo ao programa que provoca a parada da sua execução, a verificação e tratamento do referido evento e em seguida o retorno do programa ao ponto em que havia sido

interrompido.” [Pereira, 2004] Em suma, é um desvio do programa para outra tarefa e o retorno para o programa principal.

Sendo assim, quando um evento de interrupção ocorre a instrução em execução é completada, armazenando o conteúdo na pilha e o programa é desviado para o endereço no vetor de interrupção. Neste, existe uma sub-rotina que trata a interrupção, retornando ao fluxo normal do programa. [Pereira, 2004]

São ao todo 15 tipos de interrupções diferentes. Todos geram desvio do programa para o vetor de interrupção 0004h. Dessas interrupções 3 são convencionais e outras de periférico que atuam com os periféricos existentes nesse modelo de PIC. A seguir os tipos de interrupção convencionais: [Souza e Lavinia, 2005]

- **Interrupção de timer 0:** Acontece sempre que o contador TMR0 “estoura”, ou seja, sempre que ele passar de 0xFF para 0x00. Esta interrupção é utilizada para contagem de tempo.
- **Interrupção externa:** Esta é gerada por um sinal externo recebido na porta, quando configurada como entrada. Uma das finalidades dessa interrupção é reconhecer o acionamento do botão.
- **Interrupção por mudança de estado:** Essa interrupção acontece tanto na borda de subida como na borda de descida quando as portas estiverem configuradas como entradas. A mudança de estado em qualquer uma delas gera a interrupção.

4.2.2 – Conversor A/D do PIC 16F877A

Dentre os periféricos destaca-se o conversor A/D interno de 10 bits de resolução, correspondente a 1024 níveis. Este conversor pode ser configurado como interrupção e acontece quando a conversão é completada. O funcionamento do conversor A/D segue da seguinte forma:

A resolução do sinal analógico é expressa pelo seu número de bits dado pela seguinte equação, onde V_{ref} é uma tensão de referência e n é o número de bits do conversor [Souza e Lavinia, 2005]:

$$Resolução = \frac{V_{ref}}{2^n} \quad (4.1)$$

Desta forma, cada um dos n bits que compõe a informação digital representa uma parte da tensão analógica que será convertida e o total dessas partes será a tensão de entrada do conversor A/D. Assim a parte da tensão proporcional ao bit m será dada por [Souza e Lavinia, 2005]:

$$V_{entrada} = \frac{b_m 2^{(m-1)}}{2^n} V_{ref} \quad (4.2)$$

onde b_m é o valor do bit m (0 ou 1).

O sistema do conversor do PIC é conhecido como conversor de aproximação sucessiva. Nesse tipo de conversor o bit mais significativo representa a metade da tensão de referência. Desta forma, é possível saber se a tensão de entrada é maior ou menor que a metade da referência. Descobrimos o bit mais significativo segue ao próximo bit que representa a metade da metade da tensão de referência. A conversão termina chegando ao bit menos significativo [Souza e Lavinia, 2005]. Esse tipo de conversão é mais

rápido que os outros modos de conversão pois para n bits de resolução são necessários n iterações.

As tensões de referência podem ser externas, pelos pinos RA2 (V_{REF-}) e RA3 (V_{REF+}), ou internas, V_{SS} e V_{DD} . Com isso, quando a tensão de entrada for igual à V_{REF+} , a conversão resulta em 1024 e quando for igual à V_{REF-} , a conversão retorna zero. [Souza e Lavinia, 2005]

O PIC tem internamente um processo chamado *Sample and Hold*. Trata-se de um recurso que recebe uma amostra da tensão e sustenta por um período essa amostra. Este recurso funciona com o capacitor interno de 120pF que carrega a tensão de entrada. No momento em que a conversão é inicializada o capacitor é desligado do canal analógico, mantendo a tensão no capacitor constante. Isso diminui a incidência de ruídos. [Souza e Lavinia, 2005]

Capítulo 5 – Implementação

5.1 – Descrição do protótipo

O protótipo desenvolvido é um sistema que monitora a distância entre dois veículos. Além da monitoração, o sistema alerta o motorista se, a determinada velocidade V , o condutor mantém distância segura em relação ao veículo que segue a frente. Para medir a distância foi utilizado um sensor de proximidade ultra-sônico da MaxBotix EZ1. Este sensor, como citado no **capítulo 3**, tem alcance de medição de 6m. Deste modo, com apenas 6m de alcance não seria possível utilizá-lo em um veículo de tamanho real uma vez que para 40km/h seria necessário um sensor que meça no mínimo 20m dentro do método de cálculo escolhido. Para isto foi desenvolvido um modelo em escala reduzida objetivando simular o ambiente. A escala desse ambiente é de 1m real para cada 5cm do modelo. Este sensor fica fixado na parte frontal do veículo como segue na **figura 10**.



Figura 10 - Modelo de automóvel com sensor fixado a frente

O protótipo utiliza um *buzzer* que fica ligado enquanto o condutor estiver conduzindo a uma distância inferior a distância segura. Este sinalizador sonoro foi escolhido pois para alguns casos de distração, por exemplo sintonizando o rádio, o alerta sonoro é mais eficiente por não ser necessário que o motorista se mantenha atento a um ponto para constatar distância segura. Para sinalizar que o condutor mantém-se a distância mínima de segurança utiliza-se um LED que se mantém ligado enquanto o motorista estiver nessa condição. A situação é esquematizada na **figura 11**:

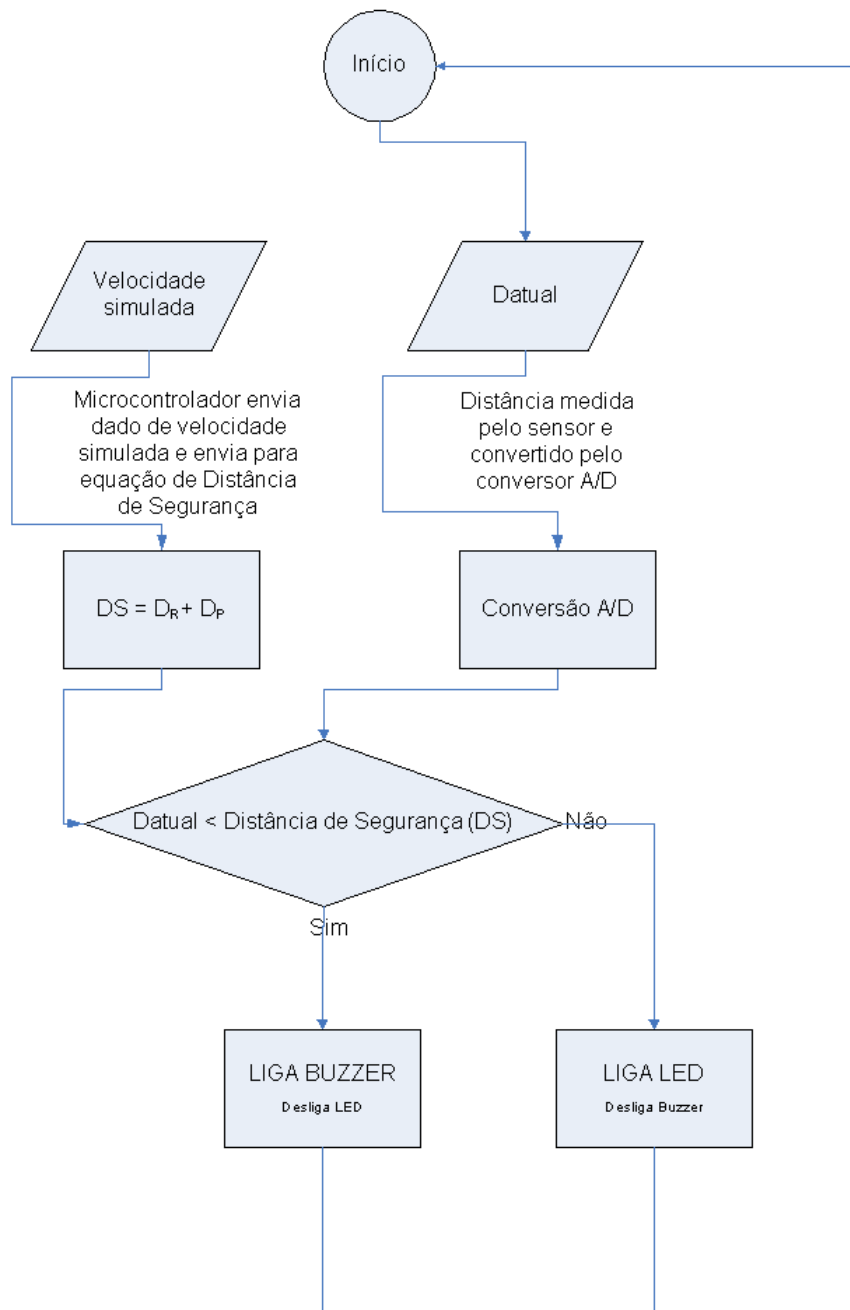


Figura 11 - Fluxograma do protótipo

O *buzzer* fica conectada no pino PA3 enquanto que o LED utilizado é o que está ligado no pino PA1 (**figura 12**).

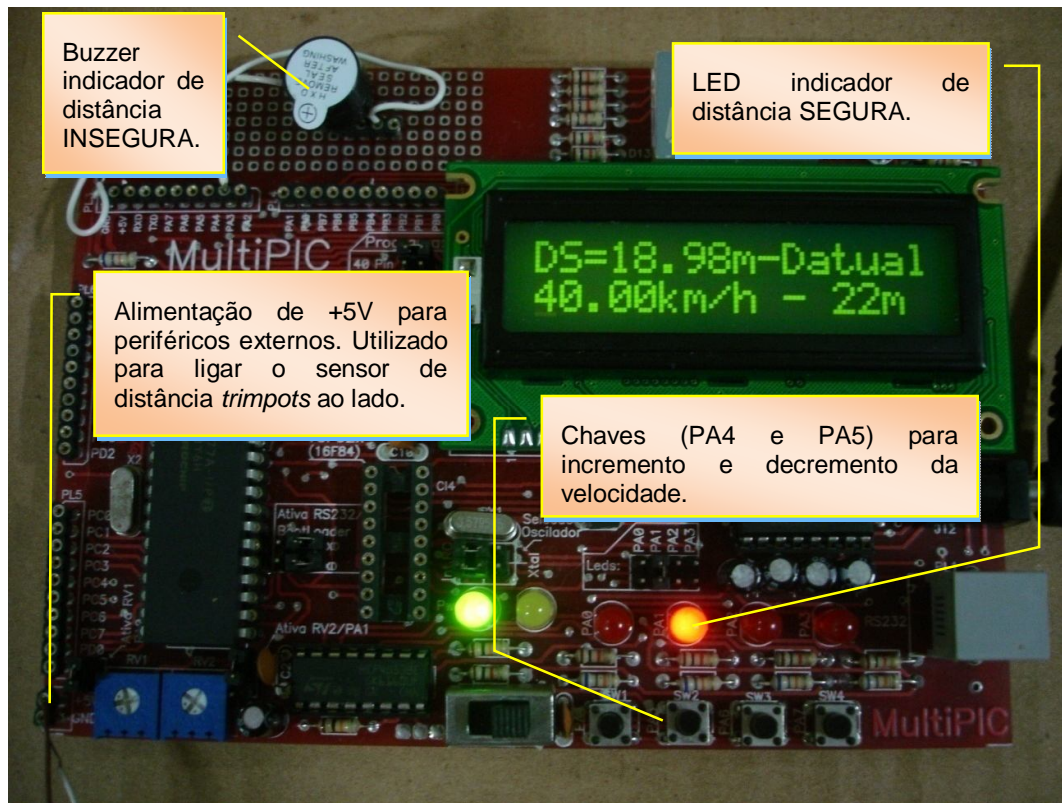


Figura 12 – MultiPIC em funcionamento e localização dos recursos

A central lógica e de controle do protótipo é feita pelo microcontrolador PIC. O sensor fica ligado em +5V pela saída de tensão independente do kit que pode ser verificado na **figura 12** no canto inferior esquerdo. Utilizando o conversor analógico/digital o sinal enviado pelo sensor é convertido e informado no *display de LCD*. Para conversão é utilizado resolução 10bits, ou seja, 1024 níveis, aproveitando o máximo suportado pelo PIC 16F877A. Foi utilizada essa resolução pois assim é possível aumentar a precisão da distância medida. Com esta resolução também foi verificado que não houve relevante queda na velocidade de conversão.

A saída analógica do sensor EZ1 (pino AN) fica conectada no pino PA0 do kit. O sensor quando ligado em +5V fornece saída de 9,8mV/pol, ou melhor, 3,92mV/cm. Isso totaliza aproximadamente 2,55V para 6,4m de leitura

e para a distância mínima de 6" (15cm) ou menor, a tensão fornecida é de 58,8mV, segundo o manual do sensor [Max Sensor, 2008].

O *LCD* é utilizado para indicar a velocidade que o veículo trafega de acordo com a simulação realizada. Nele é indicada a distância segura (DS) em metros, a distância atual (medida) em metros, "Datual", e a velocidade do veículo em quilômetros por hora identificada no primeiro valor da segunda linha. Os valores podem ser identificados na **figura 12**.

Para este ambiente a velocidade do veículo é simulada via *software* como indicação da velocidade real, já que o modelo em escala reduzida é estático. Sendo assim, para incremento e decremento da velocidade do veículo é necessária a utilização de duas chaves que estão ligadas em PA4 e PA5 vista na **figura 12**. A chave PA4 incrementa a velocidade enquanto a PA5 decrementa. Desta forma os dados de velocidade são visualizados do visor do LCD e também inseridos na equação da distância segura. No **Apêndice 2** pode ser visto o diagrama elétrico do projeto.

A escolha da simulação ao automóvel real envolve a redução de custos e uma forma mais prática para apresentação em banca. Com o modelo em escala reduzida não é necessário usar sensores de proximidades mais caros que alcancem a distância de segurança mínima para determinada velocidades, e ainda, adaptar a leitura do sensor de velocidade dos veículos automotivos, não existente nesse modelo (velocidade simulada).

A **figura 12** mostra a visão geral dos componentes utilizados no protótipo e o kit em funcionamento. Para utilização do sensor é importante desativar o *jumper* ao lado do *trimpot* RV1 para evitar sobrecarga do circuito

utilizando duas fontes de entrada para conversão analógica/digital que são a do próprio kit e a do sensor.



Figura 13 - Protótipo montado e funcionando

Na **figura 13** observa-se o protótipo montado e funcionando. Nele há a indicação da distância real de 16m e foi simulado que o veículo encontra-se a 30 km/h, o que estabelece distância de segurança de 12,76m. Nesta condição o veículo está mantendo distância segura com o obstáculo a frente dele, pois ele está acima da distância de segurança (16m) e assim o LED vermelho fica ligado.

5.2 – Ambiente de simulação do protótipo

O modelo de demonstração está sob a escala de 1m para 5cm. Esta escala foi escolhida pelo fato do sensor trabalhar com 9,8mV por polegada. O sensor que atinge até 256", ou 6,4m, são suficientes para demonstrar até 128m para o ambiente real trabalhando com essa escala. A distância de segurança de 128m vale para velocidades de até 135,9 km/h. Isto representa mais do que a legislação brasileira indica de velocidade máxima em rodovias que é de 110km/h para automóveis de passeio, quando não houver indicação de velocidade máxima da via [LEI 9503/BRASIL, 1997].

A equação utilizada para a demonstração é a de "distância de visibilidade de parada", em **2.12**. Foi escolhida essa forma de calcular pois há uma maior precisão da distância segura pelo fato de conter mais variáveis envolvidas na frenagem. Isto possibilita melhorias futuras para implementação com maior precisão dos dados, alterando-os quando for necessário e assim adequando melhor as condições de tráfego.

Nesta simulação são considerados o coeficiente de atrito do asfalto seco ($\mu = 0,80$), aceleração da gravidade de $9,8\text{m/s}^2$ e tempo de reação do motorista para acionar o freio de 1s.

Para demonstração do protótipo foi montado uma plataforma milimetrada que permite a verificação e constatação da distância medida dentro da escala estabelecida. Foi feito também um suporte para o veículo que o mantém a 30cm de altura. Com essa altura não há efeito da reflexão das ondas ultra-sônicas pela plataforma. A plataforma pode ser vista na **figura 14**.

A distância da escala foi calibrada para a escala de 5cm para 1m do automóvel real. Com isto a precisão de medição é de 5cm. Essa escolha foi levada em consideração pelo tamanho do carro em escala que mede 20cm e foi suposto que o veículo real teria 4m.

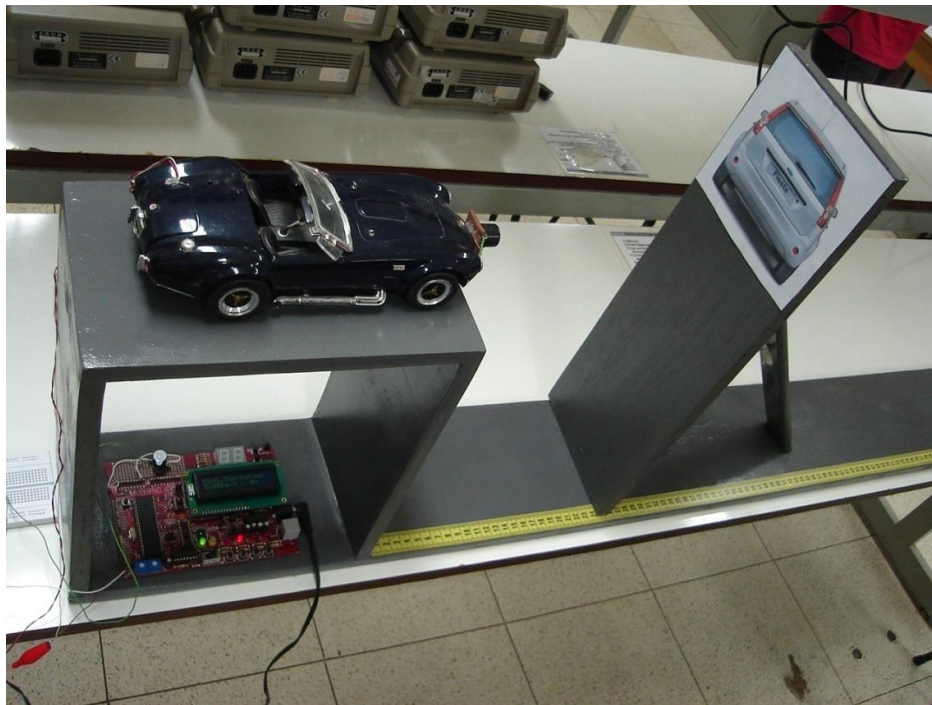


Figura 14 - Protótipo sendo testado na plataforma

5.3 – Programação do kit de desenvolvimento

Para programação do kit foi utilizado o compilador *PIC C Compiler da CCS*. Este compilador salva um arquivo em formato *.HEX. Este é o formato que o aplicativo gravador reconhece para gravar na memória do PIC. O aplicativo para gravação do código é o *IC Prog 1.06B*. [ICPROG, 2009] [CCS, 2009]

O código para este projeto foi desenvolvido em linguagem ANSI C e esta linguagem foi escolhida pela maior praticidade e pelas funções de controle

de variáveis. Além disso o autor já tinha alguma experiência com essa linguagem. Com esta linguagem é possível inserir as equações estabelecidas para a distância segura de forma mais fácil que a linguagem *Assembly*. A conversão de caracteres é feita automaticamente utilizando instruções apropriadas que convertem para os padrões de escrita que devem ser enviados ao display LCD. Assim os dados são convertidos de acordo com o padrão ASCII, sendo necessário para a correta escrita no visor de LCD. Os códigos de escrita do visor de LCD foram consultados no livro *PIC Programação em C* com algumas adaptações (MOD_LCD.C). [Pereira, 2005] Foram modificadas no algoritmo as referências dos pinos que são ligados o visor de LCD. Outro algoritmo aproveitado desse livro foi do conversor analógico/digital (exemplo 12.22) que serve para estabelecer a comunicação entre o sensor de distância e o kit. [Pereira, 2005]

A grande desvantagem de utilizar essa linguagem é quantidade de memória necessária para ter um código compilado em C. Prevendo este consumo de memória para alocar o código foi escolhido o PIC 16F877A que contém 8Kwords de memória e se mostrou suficiente para conter basicamente um código que implementa escrita no visor LCD, um conversor analógico/digital e controle de entrada e saída de dados.

5.3.1 – Principais partes do algoritmo

O algoritmo é composto de algumas funções de escrita e inicialização do LCD e uma rotina principal. Segue, nesta ordem, atribuição das variáveis para equação, a configuração do conversor A/D, selecionado PORT

A0 como entrada, utilização do *clock* interno e colocar os botões (“botao_1” e “botao_2”) em nível baixo.

Na rotina principal é feito:

O atraso de 50 μ s para carregamento do capacitor interno do conversor, leitura da conversão e atribuição da conversão a variável “valor”. Se “valor” for verdadeiro Datual recebe $(128 \times \text{valor} / 512)$. Nesta equação foi atribuído o valor máximo da distância real que seria media pelo sensor, 128m, multiplicado pelo “valor” em binário e dividido pela resolução do conversor (1024). Porém, como a tensão máxima é a metade do valor de referência do sensor (+5V), utiliza-se a metade da resolução máxima para que se iguale a tensão de referência e assim fique dentro da escala planejada.

Foi definido incremento ou decremento da velocidade de 5 em 5 km/h. Segue trecho do algoritmo abaixo:

```
5km/h) if(input(botao_2)==0){ //incremento da velocidade simulada (em
5km/h) delay_ms(300);
        vel_km+=5;
        }
        if(input(botao_1)==0){ //decremento da velocidade simulada (em
5km/h) delay_ms(300);
        vel_km-=5;
        }
```

A equação da distância segura é $DS = ((vel_km/3.6) * t_reacao)) + ((vel_km/3.6)*(vel_km/3.6)) / (2 * cx_atrito * g)$. Se “Datual” for maior ou igual a “DS”, “led” é posto em nível alto (led = 1) e “buzzer1” é posto em nível baixo

(buzzer1 = 0), do contrário põe em alto o “buzzer1” (buzzer1 = 1) e baixo “led” (led = 0). Segue trecho correspondente do algoritmo:

$$DS = ((vel_km/3.6) * t_reacao) + ((vel_km/3.6)*(vel_km/3.6)) / (2 * cx_atrito * g);$$
 // Cálculo da Distância de Segurança

//condição de distância segura

```
if(Datual>=DS){  
    output_high(led);  
    output_low(buzzer1);  
}
```

//condição de distância insegura

```
if(Datual<DS){  
    output_high(buzzer1);  
    output_low(led);  
}
```

E finalmente, escrevendo no LCD DS="X"m, Velocidade atual em "km/h" e Datual em metros. O algoritmo completo encontra-se no **Apêndice 1**.

5.4 – Dificuldades apresentadas

As dificuldades gerais do projeto foram em relação ao algoritmo e a comunicação do sensor de distância. Outro problema identificado foi na escolha da linguagem de programação. Após algumas tentativas com a programação *Assembly* foram verificadas dificuldades em estabelecer a conversão analógico/digital e com a conversão do resultado dos cálculos em caracteres padrão ASCII. Após sucessivas tentativas sem sucesso foi escolhido a linguagem C. Esta linguagem foi escolhida pois o autor já tinha algum conhecimento prévio e ainda pelo seu nível ser mais alto que o *assembly*, pois na linguagem C não é preciso utilizar métodos de conversão de

variáveis e também pela facilidade da manipulação da equação ante o *Assembly*.

5.4.1 – Dificuldades no algoritmo

Neste projeto foi verificado um problema na escrita do LCD que atrasou a conclusão deste projeto. Para enviar os caracteres para o LCD é necessário a conversão das variáveis não-caracteres em caracteres (char). A solução para este problema foi a utilização da função *printf* da linguagem C como pode ser visto no trecho do código a seguir:

```
“ lcd_pos_xy(1,1);  
  printf(lcd_escreve,"DS=%fm",DS);  
  printf(lcd_escreve,"-Datual ");  
  lcd_pos_xy(1,2);  
  printf(lcd_escreve,"%fkm/h - %lum   ",vel_km, Datual);”
```

Outro problema com o LCD foi a remoção caracteres duplicados, pois isto atrapalhava na interpretação dos dados mostrados. Para isto foi tentado utilizar a instrução de limpeza do LCD (LCD_limpa, ver algoritmo em anexo). Esta forma de limpeza não demonstrou bons resultados, pois o LCD ficou piscando intermitentemente. Outra tentativa foi utilizar essa função apenas quando os valores de distância segura e velocidade do veículo fossem alterados. Também não foi possível manter essa solução pois influenciou na medição da distância gerando imprecisão na medida. Isso ocorreu devido a perda de velocidade da conversão A/D e precisão dos dados. A medida para solucionar o problema foi o uso de espaços em branco além do que estava escrito.

Foi necessário adaptar o algoritmo do livro *PIC Programação em C* que usava como padrão o PORT B de comunicação do PIC. A correção foi feita mudando apenas a referência dos pinos. [Pereira, 2005]

O *buzzer* foi implementado utilizando o pino PA3 para ligá-lo. A tensão de funcionamento do *buzzer* é de +5V até +12V. A saída do PIC fornece +5V. A polaridade não pode ser invertida com o risco de queima do *buzzer*.

5.4.2 – Problemas com o sensor de distância e kit de desenvolvimento

O sensor foi testado com o multímetro e verificado sua estabilidade, porém quando utilizado a conversão do kit o sensor apresentava instabilidade. Essa instabilidade gerava imprecisão no alerta da distância insegura. Esse problema é comentado em um FAQ (*Frequently Asked Questions*) do sensor [MaxBotixFAQ, 2009]. Neste manual há o modo de correção. Para isso utiliza-se um capacitor de 0,1 μ F no pino do conversor A/D, pino PA0 do kit, ligado no pino de aterramento (GND). É colocado também um resistor de 10k ohms em série com a saída AN do MaxBotix EZ-1. Com isso esse circuito gera um atraso de aproximadamente 5ms permitindo que a conversão seja completada. No circuito foi utilizado um resistor de 18k ohms o que gera um retardo de 9ms, diminuindo a flutuação da medição. Este filtro cria um tipo de memória da tensão enviada pelo sensor onde o conversor faz a leitura da tensão que está no capacitor e não diretamente do sensor. As ligações elétricas estão no **Apêndice 2 – Diagrama Elétrico do Protótipo.**

Capítulo 6 – Testes e Resultados

Os testes realizados para este projeto foram: verificação da tensão de saída do sensor, verificação do comportamento do sensor com o kit e avaliação do acordo do projeto (se segue com a proposta inicial de alertar o motorista em situação de risco de colisão). Foram considerados para o teste velocidade simulada e variável, tempo de reação do motorista de 1s, coeficiente de atrito de 0,8, gravidade de $9,8\text{m/s}^2$. Foram desconsideradas diferenças da temperatura que podem influenciar no resultado da medição.

6.1 – Testes gerais do sensor

Para verificação da tensão de saída do sensor foi utilizado um multímetro comum, ligando a saída do pino AN à entrada do multímetro. Para este teste o sensor foi ligado em +5V esperando um sinal de saída de 9,8mV para cada 2,5cm. O sensor tem uma distância mínima de medição de 15cm com a tensão mínima de saída no pino AN de 58,8mV para distâncias iguais ou inferiores a 15cm, de acordo com o manual [MaxSonar, 2008]. No teste realizado a tensão mínima foi de 69mV como mostra a **figura 15**, resultando em uma diferença de 10,2mV o que equivale a 2,5cm para os 15cm medidos. Essa diferença acontece pois a saída de alimentação do kit que liga o sensor teve uma variação de 0,15V no momento deste teste. Desta forma a calibragem do sensor não foi efetuada corretamente, pois a auto-calibração do sensor recebe o valor de referência externa para selecionar a tensão máxima de saída. Ou seja, quando o sensor estiver ligado em +5V a tensão máxima no pino AN será de 2,55V. Entretanto a distância de segurança mínima é

garantida isso porque a diferença apresentada, visto na plataforma, é uma distância maior que a deveria ser medida (indicada no lcd).

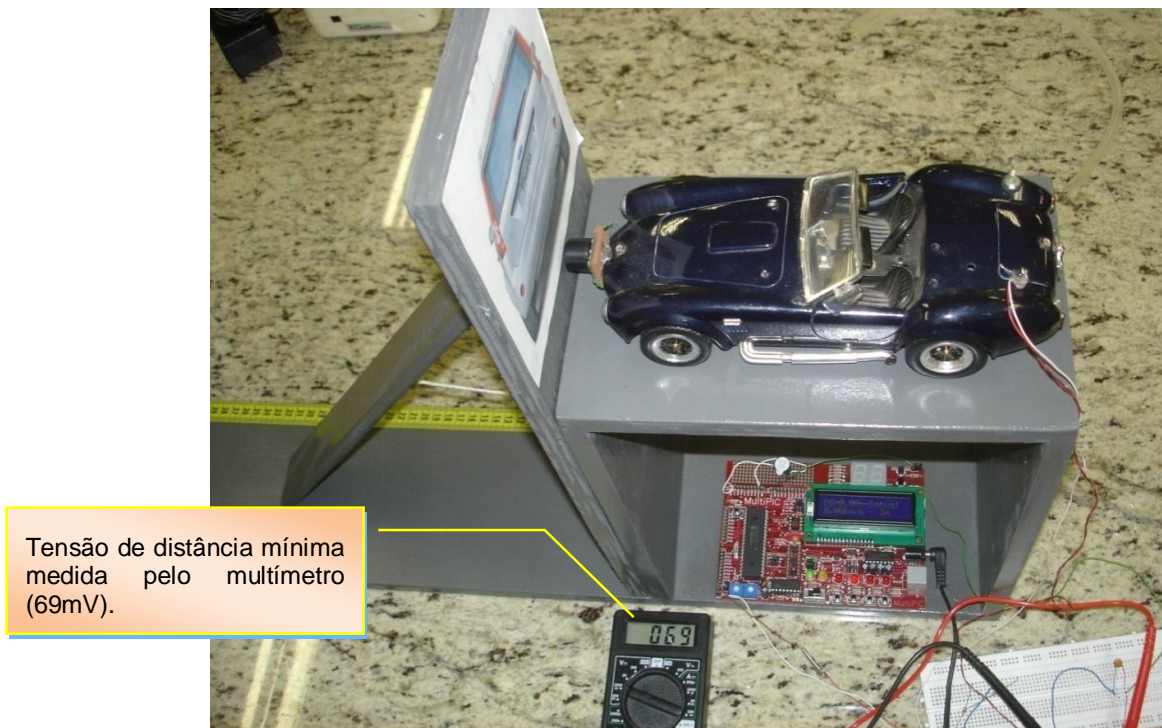


Figura 15 - Teste do sensor tensão mínima de saída no pino AN

Foi constatado que o sensor se mantém estável fazendo a leitura pelo multímetro. Já para o kit foi verificado que a leitura do sensor oscilava bastante. Isto prejudicava a precisão de medição que o protótipo deve ter pois, com essa oscilação, pode ficar alertando de forma aleatória a condição de distância segura ou insegura. De acordo com o que foi explicado em **5.4.2** foi necessário a utilização de um filtro que sustenta o tempo de amostra da tensão enviada pelo sensor, utilizando um capacitor e outros componentes. A montagem do filtro segue na **figura 16** e o esquema elétrico pode ser visto no **Apêndice 2**. Esse filtro é semelhante ao *Sample and Hold* que é embutido no

conversor A/D do PIC. Não foi possível reduzir os ruídos via *software* utilizando um tempo maior de espera do carregamento do capacitor interno do PIC.

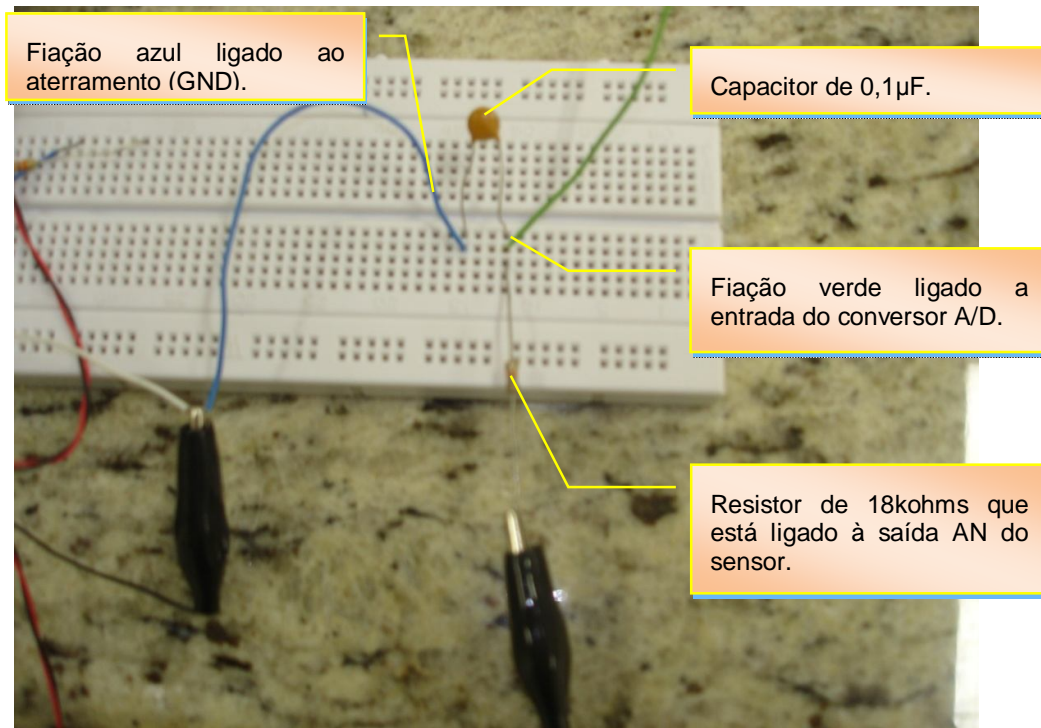


Figura 16 - Filtro para redução de ruídos do sensor

Outro fato constatado para o sensor foi o campo de sensibilidade. Como o sensor trabalha com ultra-som e o carro-modelo é baixo, 5,5cm de altura, a superfície em que este carro fica reflete as ondas ultra-sônicas “enganando” o sensor, permitindo perda de precisão da distância que o veículo se encontra. Para contornar esse problema, o carro-modelo é mantido a altura de aproximadamente 30cm ou mais da superfície de apoio para evitar que a própria superfície de apoio reflita as ondas emitidas pelo sensor. Essa distância de 30cm foi escolhida, pois foi verificado que a essa distância não havia mais efeito de reflexão do plano de apoio do carro-modelo. O suporte montado é apresentado na **figura 15**.

6.2 – Calibragem do sensor

O sensor foi calibrado para indicar 1 metro a cada 5cm reais. Sendo assim, na **figura 15**, é possível visualizar a diferença que o sensor está medindo. Esta medição mostrada na foto foi à primeira avaliação e indicou distância de 23cm para 4m quando eram esperados 20cm. As possíveis causas seriam a alimentação que está de acordo com o que foi estabelecido como referência (testado pelo multímetro 4,99V). Isso poderia ser a causa pois o sensor calibra o fator de conversão de acordo com essa tensão de alimentação na primeira vez em que é ligado. Uma má calibração pode influenciar nos resultados.

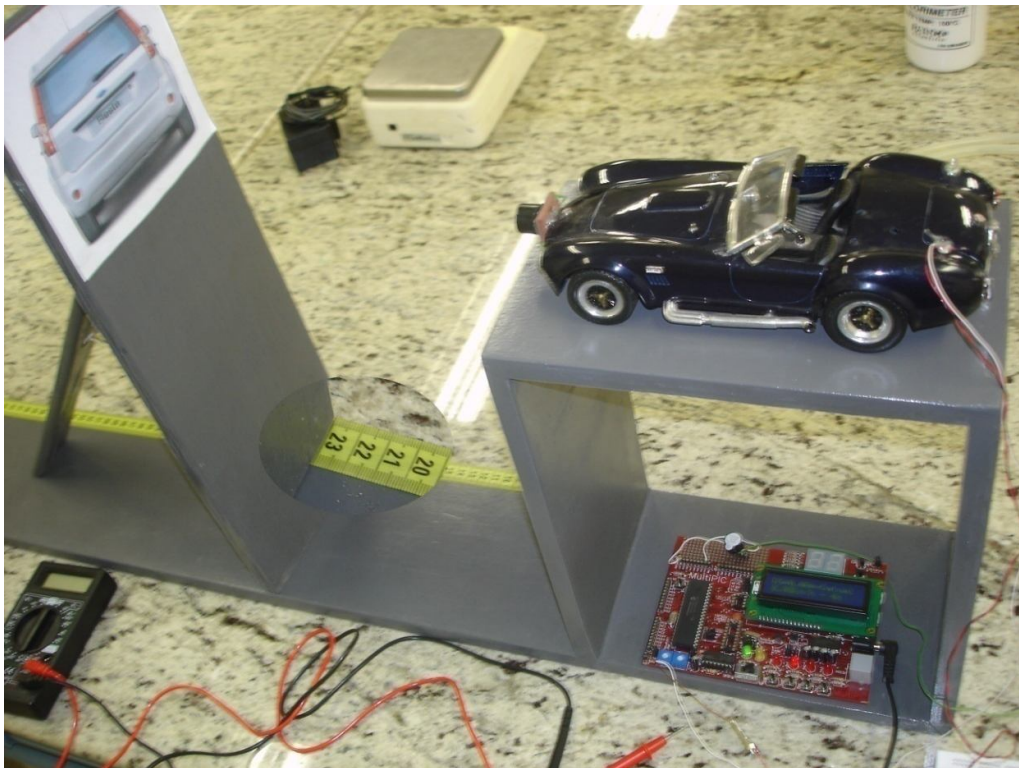


Figura 17 - Teste da distância medida com destaque para a distância medida

Outro teste realizado foi utilizando o *trimpot* do próprio kit que serve para testar o funcionamento do conversor A/D. Para utilizá-lo é necessário fechar o *jumper* RV1. Usando o *trimpot* foi possível verificar que a estabilidade da conversão é boa e não há variação da distância apresentada.

Na **tabela 9** são colocados alguns valores dos testes realizados. Foram feitos outros testes religando e calibrando o sensor. Eles indicam que o sensor não obteve a melhor calibração no primeiro teste verificado na **figura 17**:

Tabela 9 - Aferição da precisão do sensor

Distância indicada no protótipo	Distância esperada (em escala)	Distância Real (medida)
3m (distância mínima de sensibilidade do sensor)	15cm	16cm
4m	20cm	21cm
5m	25cm	27cm
6m	30cm	32cm
25m (distância máxima da plataforma)	125cm	132cm

A diferença do primeiro teste para os demais é registrada na **tabela 9** e indica que a diferença caiu aproximando do valor estabelecido.

Este sensor pode sofrer variações na medição dependendo da temperatura do ar pois não há nenhum algoritmo de compensação. [MaxSonar, 2008] Não foi considerado também fenômenos meteorológicos, tais como chuva e neblina, que podem alterar o resultado da medição devido a alteração da velocidade do ultra-som nesses ambientes.

6.3 – Teste do alerta de distância segura

O teste do alerta de distância segura foi feito simulando a velocidade utilizando os botões de incremento e decremento (PA1 e PA2). Com isso a distância é calculada, indicada no LCD e variando a distância do anteparo é possível verificar se o alerta funciona de acordo com o que foi estabelecido que é ligar o *buzzer* e desligar o LED vermelho. O anteparo simula o veículo que estaria em trânsito na frente deste. Na **figura 18** é evidenciada a seguinte situação:

O motorista segue com seu veículo a 15km/h. O alerta identifica distância de segurança é de 5,27m para esta velocidade. Porém o motorista mantém distância medida pelo sensor de 4m do suposto carro da frente. Essa situação indica condição de alerta e risco de não evitar um acidente visto que o motorista não mantém distância de segurança do veículo da frente. Desta forma o monitor alerta para o motorista, por meio de um *buzzer* e visualmente por meio de um LED vermelho que fica desligado, que ele está abaixo da distância de segurança para velocidade de 15km/h e este deve afastar-se do veículo da frente.

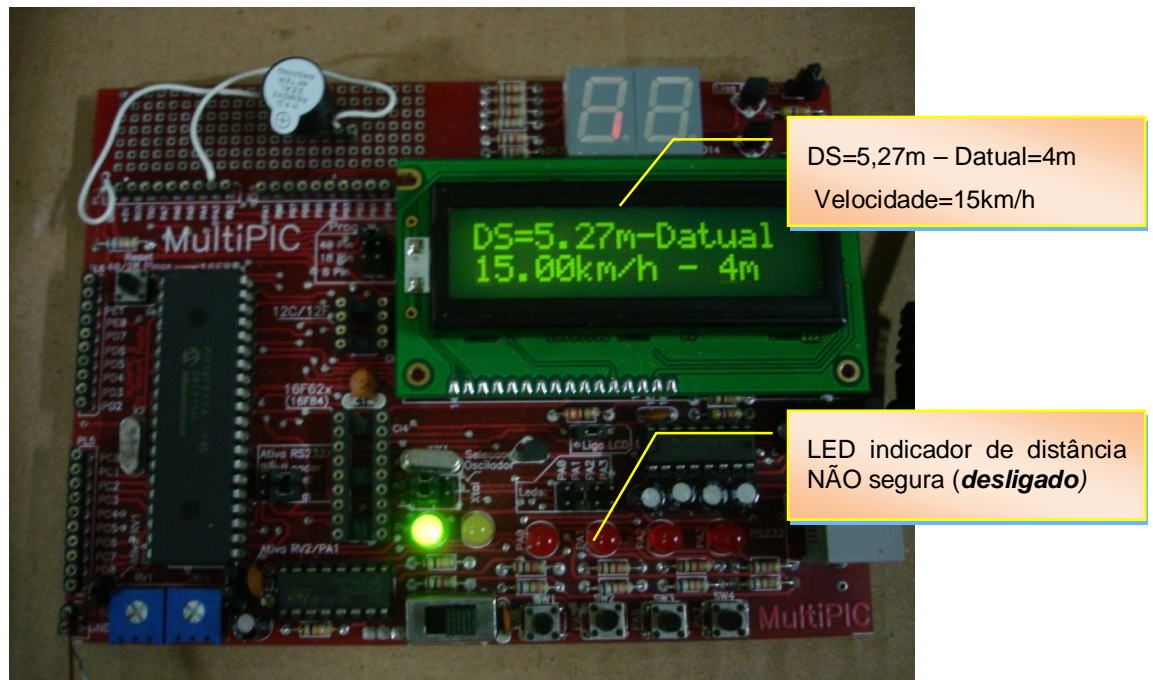


Figura 18 - Teste indicando que a distância é insegura

Os testes confirmam a proposta do projeto de alertar o motorista para distância insegura em trânsito. Foi observado que utilizando a saída analógica há perda de precisão devido à discretização que o método de conversão trabalha. Isto é, no intervalo entre dois níveis o microcontrolador tenta fazer a aproximação do ponto realmente medido e desta forma, em alguns momentos, há pequena oscilação do resultado. Essa situação ocorre devido à limitação da resolução do conversor analógico deste modelo de microcontrolador, que é de 10bits.

Capítulo 7 – Conclusão

O projeto proposto permitiu avaliar alguns métodos de cálculo de distância de segurança que se deve manter do veículo da frente e ver o que melhor se adapta as mais variadas condições de trânsito. Além disso foi possível colocar em prática alguns conhecimentos adquiridos neste curso de Engenharia para a elaboração de um protótipo, bem como avaliar as dificuldades apresentadas e resolvê-las.

7.1 – Conclusões gerais

Neste projeto foram verificadas as dificuldades que ocorrem em projetos de Engenharia bem como lidar com essas situações. Foram aplicados alguns conhecimentos adquiridos ao longo desses anos e pesquisar outros novos que são necessários para superar as dificuldades. A finalidade deste projeto foi atingida utilizando aquilo que foi proposto: o sensor de distância e um microcontrolador. O alerta de segurança é acionado de acordo com o que foi estabelecido. Este monitor de distância de segurança facilita verificar a distância de segurança para o motorista e alertá-lo se ele mantém distância segura ou não.

Foram relevantes as disciplinas de Física que auxiliaram na compreensão da dinâmica de frenagem dos veículos; Microcontroladores e Microprocessadores que foi a base para o conhecimento de programação e funcionamento do microcontrolador; Linguagem e Técnica de Programação que serviu de base para a programação na linguagem C; e Circuitos Eletrônicos que foi o auxílio para a montagem correta dos componentes.

7.2 – Conclusões dos métodos de distância segura

Foram avaliados 3 métodos de determinação da distância de segurança e foi concluído que o método chamado de Distância de Visibilidade de Parada mostrou-se melhor diante dos demais devido, a maior quantidade de variáveis, (como por exemplo a velocidade, tempo de reação do motorista, coeficiente de atrito), permitindo implementar e melhorar a precisão da distância de segurança. Neste método foi verificado que a **equação 2.10** indica a distância de frenagem próxima do valor real obtido em testes de revistas especializadas e evidenciado na tabela 6. A “regra dos 2s”, embora menos precisa, pode ser utilizada e é ensinada por várias instituições de trânsito no mundo. Essa regra atende bem velocidades de até 60km/h. Essa regra, apesar de ter um método fácil de verificação descrito no **capítulo 2**, pode acarretar que o motorista se distraia e conseqüentemente cause um acidente. Isto evidencia que um recurso tecnológico auxilia na segurança do condutor e dos passageiros. Além da segurança para os passageiros o motorista evita depreciação do seu patrimônio e o Estado diminui os gastos de socorro médico em acidentes de trânsito.

7.3 – Conclusões de aspecto técnico

No aspecto técnico, pôde-se verificar que o sensor utilizado tem relativa precisão dos dados porém não é tão robusto, podendo sofrer interferências com o meio externo. Uma dessas interferências é o raio de alcance do sensor ultra-sônico, que se estende longitudinalmente refletindo em objetos próximos resultando em medições erradas. A distância máxima de medição, (6m), limita sua utilização em carros reais visto que a distância de

segurança para 40km/h é de pelo menos 19,5m de acordo com os cálculos. A saída analógica foi uma boa alternativa para utilização nesta simulação e atendeu as expectativas até certo ponto bem, visto o resultado da simulação realizada, haja vista que se não houver uma fonte de alimentação estabilizada o valor de referência altera e a precisão do monitor também. A precisão da distância aferida foi em média de 5% utilizando o conversor A/D interno do PIC com sua máxima resolução. Este sensor foi o melhor com custo X benefício para utilização no meio acadêmico pois ele custa 3,96% de um modelo industrial (da SICK®) com características semelhantes (ver **3.2.2**).

O microcontrolador PIC16F877A não perde desempenho em suas tarefas dentro deste projeto mesmo utilizando a máxima resolução do conversor A/D. A adaptação em automóveis reais requer um sensor mais robusto e de maior alcance de medição e, também, adaptar o sensor de velocidade do carro para se comunicar com o monitor. Um sensor que meça até 60m serve bem para verificar distância segura para velocidades de até 80km/h, onde são necessários 53,6m de distância de segurança pela regra da distância de visibilidade de parada. Desta forma atenderia bem a maior parte das situações onde a maior concentração de automóveis está nos centros urbanos.

7.4 – Propostas de trabalhos futuros

Sugerem-se algumas formas de avaliar a distância de segurança e alertar o motorista se ele mantém essa condição. No entanto a simulação apresentada neste projeto previu apenas uma situação estática com pista seca,

tempo de reação do motorista de 1s simulando apenas a velocidade. Para melhorias futuras seria interessante:

1. Pesquisar outros sensores de distância que meçam maiores distâncias, sejam mais estáveis e tenham compensação de temperatura, possibilitando sua utilização em automóveis;
2. Melhorar a identificação de outras variáveis além da velocidade:
 - a. Coeficiente de atrito da pista (seca, molhada, etc);
 - b. Tempo de reação do motorista (durante a noite ou dia, pela idade);
3. Estudar a possibilidade de manter o veículo automaticamente na distância segura;
4. Frear automaticamente em iminência de colisão;
5. Construir um protótipo de simulação dinâmica.

Referência Bibliográfica

16F877XA, Microchip Data Sheet. Disponível em:
<<http://ww1.microchip.com/downloads/en/DeviceDoc/39582b.pdf>>. Acessado
em outubro de 2008.

AAA Foundation - Artigo – **Distractions In Everyday Driving**-
<<http://www.aaafoundation.org/pdf/DistractionsInEverydayDrivingHiRes.pdf>>
Capítulo de Introdução - Acessado em: setembro de 2008.

ARAGAO, Ranvier Feitosa - Acidentes De Trânsito: Aspectos
Técnicos e Jurídicos Campinas : Millenium, 2003 , 3.ed

BOSCH - Freie com maior segurança com o Antilock Braking
System ABS (sistema antibloqueio de frenagem) - [http://rb-
kwin.bosch.com/br/pt/safety_comfort/drivingsafety/abs/brakemoresafely.html](http://rb-kwin.bosch.com/br/pt/safety_comfort/drivingsafety/abs/brakemoresafely.html)
Acessado em setembro de 2008.

BRASIL, LEI 9.503 /. Presidência da República – Código de Trânsito
Brasileiro - Acessado em: Agosto de 2008.
<<http://www.planalto.gov.br/ccivil/leis/L9503.htm>>.

CCS, PIC C Compiler PCWH, acessado em Abril 2009,
<http://www.ccsinfo.com/downloads.php>

DENATRAN e Fundação Carlos Chagas. Manual Básico de
Segurança no Trânsito. - Cap. 4 -
<http://www.anfavea.com.br/documentos/indicesseguranca.pdf>. - Acessado em
Agosto de 2008

DISTRONIC, Drive Disk. 2008. Disk Drive - Distance Control System
- http://www.diskdrive.co.za/newsstories/mercedes_distronic.html. Acessado em: Agosto de 2008

DOTTA, Atico J. **O** Condutor Defensivo: Teoria E Pratica - Porto Alegre : Sagra Luzzato , 1998 , 2.ed

G1.com - Abertura de capô causa acidente em SP -
<<http://g1.globo.com/Noticias/SaoPaulo/0,,MUL183991-5605,00.html>> - Acesso em setembro de 2008

ICPROG, Programador de Protótipos, Acessado em Abril 2009,
<http://www.ic-prog.com/icprog106B.zip>

Kilsztajn, Samuel; Silva, César Roberto Leite da; Silva, Dorivaldo Francisco da; Michelin, André da Cunha; Carvalho, Aissa Rendall de; Ferraz, Ivan Lopes Bezerra: - Artigo: Taxa de mortalidade por acidentes de trânsito e frota de veículos – 2001 -
http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0034-89102001000300008 – Acessado em setembro de 2008

Max Sonar – Manual do sensor MaxBotix EZ1 -
<http://www.tato.ind.br/files/EZ1.pdf> - Acessado em Setembro de 2008.

MaxBotix FAQ – Frequently Asked Questions do sensor MaxBotix EZ1 -
http://www.maxbotix.com/MaxSonar-EZ1_FAQ.html#New_Flash_UAV_and_Mobile_Robotic_Users Acessado em Março de 2009.

NEGRINI NETO, Osvaldo. 2003. *Dinâmica dos Acidentes de Trânsito: Análises e Reconstruções*. Campinas : Millenium, 2003.

NUSSENZVEIG, Herch Moyses. 2002. *CURSO DE FÍSICA BÁSICA: MECANICA*. SÃO PAULO : EDGARD BLUCHER, 2002.

PEREIRA, Fábio. Microcontroladores PIC – Programação em C. 4ª ed. São Paulo: Érica, 2005.

PEREIRA, Fábio. Microcontroladores PIC – Técnicas Avançadas. 3ª ed. São Paulo: Érica, 2004.

QUATRO RODAS, Revista – Ano 47 – Edição 567 – Julho de 2007.

ROSARIO, Joao Mauricio – Princípios de Mecatrônica São Paulo : Pearson , 2005

ROSPA, The Royal Society for the Prevention of Accidents - Driver Distraction -

http://www.rospace.com/roadsafety/advice/driving/info/driver_distraction.pdf -

Acessado em Fevereiro de 2009

ROSPA, The Royal Society for the Prevention of Accidents - Recognise what makes you speed - keeping up with traffic, overtaking or being tailgated - <http://www.rospace.com/roadsafety/toptentips/speedtriggers.htm>

Acessado em Setembro de 2008

SICK, Sensor Intelligence -

<http://www.sick.com.br/br/produtos/sensoresindustriais/sensoresultrasom/pt.html> - Acessado em setembro de 2008.

SMARTRADIO, soluções e sistemas especiais para RadioComunicação Profissional, <http://www.smartradio.com.br/website/index.html>, acessado em setembro de 2008.

SOUZA, David José; **LAVINIA**, Nicolás César. Conectando o PIC 16F877A – Recursos Avançados. 2ª ed. São Paulo: Érica, 2005.

TATO – Tato Equipamentos Eletrônicos - <http://www.tato.ind.br/> - Acessado em Setembro de 2008.

VISUAL EXPERT - "*How Long Does It Take To Stop? Methodological Analysis of Driver Perception-Brake Times*" Transportation Human Factors, 2, pp 195-216, 2000. - <http://www.visualexpert.com/Resources/reactiontime.html> - Acessado em Novembro de 2008.

Apêndice 1 – Algoritmo do Projeto

```
/*  
Uniceub - FATECS  
Projeto Final - Engenharia de Computação -  
Gustavo e Silva Maciel - RA:20218190  
Monitor de distância de segurança para automóveis utilizando  
sensor de distância.
```

```
Escreve no LCD Distância Segura, Distância atual e velocidade  
atual.
```

```
Acende LED na porta PA1 se Distância Atual for maior ou igual a  
Distância Segura.
```

```
Liga Buzzer ligado na porta PA3 se Distância Atual for menor que  
Distância Segura.
```

```
Sensor de distância ligado na porta PA0, porta onde fica o conversor  
A/D.
```

```
Resolução de 10bits
```

```
*/
```

```
#include <16f877A.h>  
#device adc=10  
#include <stdio.h>  
#include <math.h>  
#use delay(clock=4000000)  
#fuses XT,NOWDT,PUT,NOBROWNOUT,NOLVP  
#ifndef lcd_enable  
#define lcd_enable    pin_B3    // pino enable do LCD  
#define lcd_rs        pin_B5    // pino rs do LCD  
//#define lcd_rw        pin_e2    // pino rw do LCD  
#define lcd_d4        pin_B0    // pino de dados B0 do LCD  
#define lcd_d5        pin_B1    // pino de dados B1 do LCD  
#define lcd_d6        pin_B2    // pino de dados B2 do LCD  
#define lcd_d7        pin_B4    // pino de dados B4 do LCD  
#endif  
#define lcd_type 2        // 0=5x7, 1=5x10, 2=2 linhas
```

LCD

```
#define lcd_seg_lin 0x40 // Endereço da segunda linha na RAM do
```

```
#define botao_1      pin_A4 //Decremento da velocidade
```

```
#define botao_2      pin_A5 //Incremento da velocidade
```

```
#define buzzer       pin_A2 //Saída para ligar o Buzzer
```

```
#define buzzer1      pin_A3 //Saída para ligar o Buzzer
```

```
#define led          pin_A1 //Saída para ligar o led
```

LCD

```
// a constante abaixo define a seqüência de inicialização do módulo
```

```
byte CONST INI_LCD[4] = {0x20 | (lcd_type << 2), 0xf, 1, 6};
```

```
byte lcd_le_byte()
```

```
// lê um byte do LCD (somente com pino RW)
```

```
{
```

```
    byte dado;
```

```
    // configura os pinos de dados como entradas
```

```
    input(lcd_d4);
```

```
    input(lcd_d5);
```

```
    input(lcd_d6);
```

```
    input(lcd_d7);
```

```
    // se o pino rw for utilizado, coloca em 1
```

```
    #ifdef lcd_rw
```

```
        output_high(lcd_rw);
```

```
    #endif
```

```
    output_high(lcd_enable); // habilita display
```

```
    dado = 0; // zera a variável de leitura
```

```
    // lê os quatro bits mais significativos
```

```
    if (input(lcd_d7)) bit_set(dado,7);
```

```
    if (input(lcd_d6)) bit_set(dado,6);
```

```
    if (input(lcd_d5)) bit_set(dado,5);
```

```
    if (input(lcd_d4)) bit_set(dado,4);
```

```
    // dá um pulso na linha enable
```

```
    output_low(lcd_enable);
```

```

    output_high(lcd_enable);
    // lê os quatro bits menos significativos
    if (input(lcd_d7)) bit_set(dado,3);
    if (input(lcd_d6)) bit_set(dado,2);
    if (input(lcd_d5)) bit_set(dado,1);
    if (input(lcd_d4)) bit_set(dado,0);
    output_low(lcd_enable); // desabilita o display
    return dado; // retorna o byte lido
}

```

```

void lcd_envia_nibble( byte dado )
// envia um dado de quatro bits para o display
{
    // coloca os quatro bits nas saidas
    output_bit(lcd_d4,bit_test(dado,0));
    output_bit(lcd_d5,bit_test(dado,1));
    output_bit(lcd_d6,bit_test(dado,2));
    output_bit(lcd_d7,bit_test(dado,3));
    // dá um pulso na linha enable
    output_high(lcd_enable);
    output_low(lcd_enable);
}

```

```

void lcd_envia_byte( boolean endereco, byte dado )
{
    // coloca a linha rs em 0
    output_low(lcd_rs);
    // aguarda o display ficar desocupado
    //while ( bit_test(lcd_le_byte(),7) ) ;
    // configura a linha rs dependendo do modo selecionado
    output_bit(lcd_rs,endereco);
    delay_us(100); // aguarda 100 us
    // caso a linha rw esteja definida, coloca em 0

```



```

#ifdef lcd_rw
    output_low(lcd_rw);
#endif
// desativa linha enable
output_low(lcd_enable);
// envia a primeira parte do byte
lcd_envia_nibble(dado >> 4);
// envia a segunda parte do byte
lcd_envia_nibble(dado & 0x0f);
}

void lcd_ini()
// rotina de inicialização do display
{
    byte conta;
    output_low(lcd_d4);
    output_low(lcd_d5);
    output_low(lcd_d6);
    output_low(lcd_d7);
    output_low(lcd_rs);
#ifdef lcd_rw
    output_high(lcd_rw);
#endif
    output_low(lcd_enable);
    delay_ms(15);
    // envia uma seqüência de 3 vezes 0x03
    // e depois 0x02 para configurar o módulo
    // para modo de 4 bits
    for(conta=1;conta<=3;++conta)
    {
        lcd_envia_nibble(3);
        delay_ms(5);
    }
}

```

```

    lcd_envia_nibble(2);
    // envia string de inicialização do display
    for(conta=0;conta<=3;++conta) lcd_envia_byte(0,INI_LCD[conta]);
}

```

```

void lcd_pos_xy(byte x, byte y)
{
    byte endereco;
    if(y!=1)
        endereco = lcd_seg_lin;
    else
        endereco = 0;
    endereco += x-1;
    lcd_envia_byte(0,0x80|endereco);
}

```

```

void lcd_escreve( char c)
// envia caractere para o display
{
    switch (c)
    {
        case '\f' : lcd_envia_byte(0,1);
                    delay_ms(2);
                    break;
        case '\n' :
        case '\r' : lcd_pos_xy(1,2);
                    break;
        case '\b' : lcd_envia_byte(0,0x10);
                    break;
        default  : lcd_envia_byte(1,c);
                    break;
    }
}

```

```

main()
{

    /*DS (Distância de Segurança),
    cx_atrito (coeficiente de atrito),
    g (gravidade),
    vel_km (velocidade do veículo),
    t_reação (tempo de reação)*/

    float DS, cx_atrito, g, vel_km, t_reacao;
    int32 Datual;
    int32 valor;

    lcd_ini(); // inicializa LCD
    setup_ADC_ports(RA0_analog); //configuração do conversor A/D
    setup_adc(ADC_CLOCK_INTERNAL);
    set_adc_channel(0);
    output_low(botao_1);
    output_low(botao_2);

    cx_atrito=0.8;
    g=9.8;
    vel_km=5;
    t_reacao=1;

    lcd_ini();
    lcd_pos_xy(1,1);
    printf(lcd_escreve," Uniceub ");
    lcd_pos_xy(1,2);
    printf(lcd_escreve," Gustavo Maciel ");
    delay_ms(2500);
    lcd_escreve("\f");
    lcd_pos_xy(1,1);

```

```

    printf(lcd_escreve,"  FATECS  ");
lcd_pos_xy(1,2);
    printf(lcd_escreve,"Eng de Computacao");
    delay_ms(2500);
    lcd_escreve("\f");
lcd_pos_xy(1,1);
    printf(lcd_escreve,"Monitor de  ");
lcd_pos_xy(1,2);
    printf(lcd_escreve,"Distância Segura ");
    delay_ms(2500);
    lcd_escreve("\f");
while (true)
{
    delay_us(50);
    valor = read_adc(); // efetua conversão A/D
    if(valor)
        Datual = (128*valor) / 512; // "Datual" recebe o valor em metros
    if(input(botao_2)==0){ //incremento da velocidade simulada (em
5km/h)
        delay_ms(300);
        vel_kmh+=5;
    }
    if(input(botao_1)==0){ //decremento da velocidade simulada (em
5km/h)
        delay_ms(300);
        vel_kmh-=5;
    }

    DS = ((vel_kmh/3.6) * t_reacao) + ((vel_kmh/3.6)*(vel_kmh/3.6)) / (2
* cx_atrito * g); // Cálculo da Distância de Segurança

    //condição de distância segura
    if(Datual>=DS){
        output_high(led);
        output_low(buzzer1);
    }
}

```

```

    }

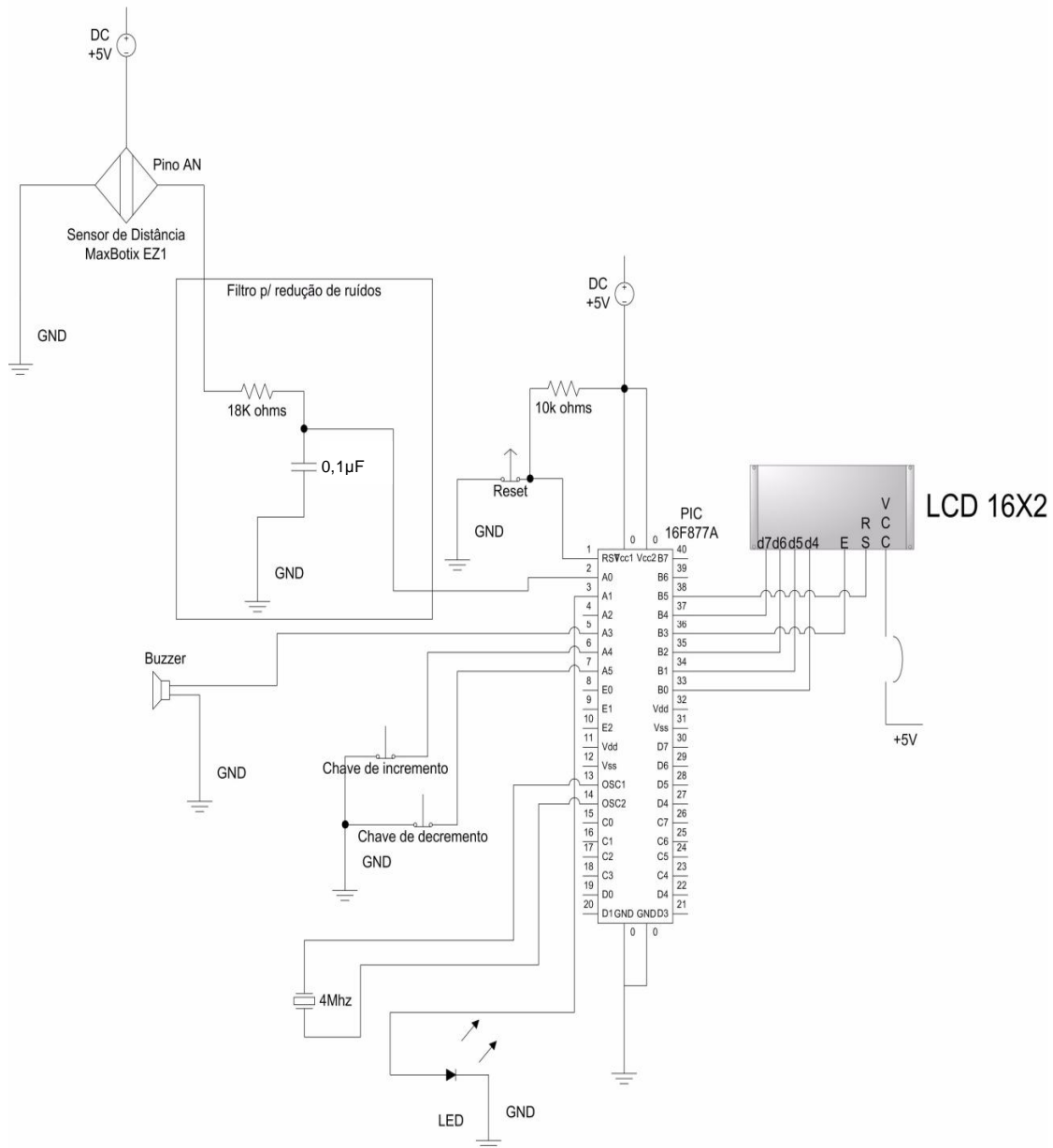
    //condição de distância insegura
    if(Datual<DS){
        output_high(buzzer1);
        output_low(led);
    }

    //escrita de DS, Datual e Velocidade no LCD
    lcd_pos_xy(1,1);
    printf(lcd_escreve,"DS=%fm",DS);
    printf(lcd_escreve,"-Datual ");
    lcd_pos_xy(1,2);
    printf(lcd_escreve,"%fkm/h - %lum   ",vel_kmh, Datual);

}
}

```

Apêndice 2 – Diagrama elétrico do protótipo



Comentários:
 -Chave PA4 incrementa velocidade simulada
 -Chave PA5 decrementa velocidade simulada
 -LED de aviso de distância de segurança OK

Apêndice 3 – Custo do protótipo

1. Kit de desenvolvimento MultiPIC –	R\$ 200,00
2. Sensor Ultra-sônico MaxBotix EZ1 –	R\$ 99,00
3. Buzzer –	R\$ 5,00
4. Plataforma em madeira –	R\$ 150,00
5. Carro em escala –	R\$ 20,00
Total –	R\$ 474,00